ELSEVIER

End-to-end quality of service provisioning through inter-provider traffic engineering

Michael P. Howarth[a,*], Mohamed Boucadair[b], Paris Flegkas[a], Ning Wang[a], George Pavlou[a], Pierrick Morand[b], Thibaut Coadic[b], David Griffin[c], Abolghasem Asgari[d], Panos Georgatsos[e]

[a]*University of Surrey, Guildford GU2 7XH, UK*
[b]*France Telecom R&D, Rue des Coutures, 14066 Caen, France*
[c]*University College London, London WC1E 7JE, UK*
[d]*Thales Research and Technology Ltd., Worton Drive, Worton Grange, Reading RG2 0SB, UK*
[e]*Algonet SA, 17672 Kalithea, Athens, Greece*

Abstract

This paper addresses the issue of delivering solutions that will enable the incremental implementation of inter-domain quality of service (QoS) in the multi-provider commercial Internet. The paper first introduces a holistic architecture that describes the key functions required to support inter-domain QoS, and then proceeds to present results from two major components of the architecture. A genetic algorithm for QoS-aware offline inter-domain traffic engineering is first presented, and it is shown through simulation studies how this can optimise the apportionment of QoS provisioning between adjacent domains. Secondly, QoS enhancements to BGP are proposed and the results of a testbed implementation are described, demonstrating how this QoS-enhanced BGP can deliver inter-domain QoS routing.
© 2005 Elsevier B.V. All rights reserved.

Keywords: QoS; Inter-domain; Traffic engineering; BGP; Service level specification; SLS

1. Introduction

The demands placed on the Internet continue to increase, with increased development of multimedia applications and distributed data retrieval systems. It is thus necessary to extend the current best-effort Internet to support Quality of Service (QoS) to satisfy the end-to-end performance requirements of these applications. Deployable architectures that support QoS are recognised as an important step for the next-generation Internet [1], and QoS-based services are seen as a part of future Internet service offerings [2].

Most research to date has focused on supporting QoS within a single administrative domain. However, delivery of end-to-end QoS to support end-user applications requires that domains administered by different organisations cooperate in order to deliver the required level of service. The problem we therefore seek to address is how to provide QoS across multiple domains in a way that takes into account the commercial Internet's multi-organisational structure, builds incrementally on existing protocols and approaches, and most importantly, is scalable. This is the task that has been addressed by the EU IST MESCAL (Management of End to end quality of ServiCe Across the internet at Large) project, and in this paper we describe the MESCAL approach. We present first a functional architecture that describes the functions required to support inter-domain QoS, and then describe a set of three different solutions that implement these functions. We subsequently present results obtained from inter-domain traffic engineering (TE) simulation work, as well as results from testbed-based experiments.

1.1. Related work

A significant body of research work exists in the literature on intra-domain QoS solutions. A QoS-based

* Corresponding author. Tel: +44 1483 683034; fax: +44 1483 686011
E-mail addresses: m.howarth@surrey.ac.uk (M.P. Howarth), mohamed.boucadair@francetelecom.com (M. Boucadair), p.flegkas@surrey.ac.uk (P. Flegkas), n.wang@surrey.ac.uk (N. Wang), g.pavlou@surrey.ac.uk (G. Pavlou), pierrick.morand@francetelecom.com (P. Morand), thibaut.coadic@francetelecom.com (T. Coadic), dgriffin@ee.ucl.ac.uk (D. Griffin), hamid.asgari@thalesgroup.com (A. Asgari), pgeorgat@egreta.gr (P. Georgatsos).

intra-domain network architecture and intra-domain service level specification (SLS) were defined in [3,4], respectively. A lot of research work has also addressed intra-domain traffic engineering as an effective paradigm to achieve edge-to-edge QoS. Among these solutions, many authors use Multi Protocol Label Switching (MPLS) to support explicit routing (MPLS-based TE) [5,6]. More recently, approaches to control traffic by intelligently setting the link weights of link-state routing protocols such as OSPF and IS–IS have also been proposed (IP-based TE) [7,8].

In contrast to the intra-domain scenario, limited work has been conducted on inter-domain QoS. Many authors have considered inter-domain traffic engineering for best effort traffic: for example, [9] has proposed methods for effectively tuning inbound/outbound traffic across multiple domains using Border Gateway Protocol (BGP), and [10] considered efficient TE negotiations between adjacent ISPs. However, this work on plain BGP does not take QoS into consideration. Other research work supports end-to-end QoS for inter-domain traffic. A number of key components for an inter-domain QoS architecture were described in [11]; however the work we present here is we believe the first to provide a complete description of the functionality required to fully support inter-domain QoS. In [12] the authors also proposed a heuristic algorithm for inter-domain traffic engineering with QoS constraints. Ref. [13] proposed service level agreement (SLA) policies to enable IP network providers (INPs) to agree how to distribute service-level constraints (i.e. QoS) across multiple domains. Other proposals have addressed QoS extensions to the underlying BGP [14,15] to enable QoS dissemination and routing across multiple autonomous systems (ASs). However, it should be noted that all these contributions are ad hoc solutions without an integrated or systematic view of how to deploy Internet QoS in a scalable and incremental fashion.

1.2. Structure of the paper

The rest of this paper is organised as follows. Section 2 presents a vocabulary for inter-domain QoS models. In Section 3 we outline the functional architecture that includes the key functions required to provide inter-domain QoS across the multi-provider commercial Internet. Section 4 specifies three solution options for inter-domain QoS delivery, each of which is a customisation of algorithms within the architecture described in Section 3. The following sections describe two key components for achieving end-to-end QoS that implement particular algorithms within the architecture introduced in Section 3. In Section 5, we present algorithms and simulation results for offline inter-domain QoS-enabled traffic engineering. In Section 6, we propose a set of enhancements to BGP to enable it to support QoS requirements, and in Section 7, we describe a testbed-based implementation of QoS-enabled BGP (q-BGP). Finally, in Section 8 we present a summary and draw our conclusions.

2. The MESCAL QoS models

This Section defines the key terms used in the MESCAL architecture, in order to establish a vocabulary for inter-domain QoS models. We first introduce the business entities that are the principal actors in inter-domain QoS and then define the QoS-based technical language, drawing on and extending the intra-domain QoS model developed and validated in the TEQUILA project [3,16].

2.1. Business model

Fig. 1 illustrates the business relationships between the principal entities assumed in MESCAL. A *Customer* subscribes to QoS-based services offered by Providers. Customers are the target recipients of QoS-based services offered on the basis of SLAs. The provider that is the focus of MESCAL is the INP, since we are concerned with QoS-based IP connectivity services, i.e. services that provide reachability with particular QoS between hosts in the IP address space. An INP owns and administers an IP network infrastructure; in this paper for simplicity we assume that this IP network comprises a single domain or autonomous system (AS).

The focus of the MESCAL project is the business relationships between Customers and INPs, and also between INPs, for the purpose of expanding the geographic space of their QoS-based IP connectivity services. This business relationship is represented by SLAs and more specifically by their technical aspects, the SLS. In the rest of this paper the term provider will be used to mean an INP unless otherwise stated.

2.2. QoS-based services

Services that provide QoS-based value to customers are offered on the basis of SLAs. The latter are established between customers and providers, and describe the services offered, their characteristics and the duties of customers in using the services. The SLS is an integral part of a SLA, denoting the technical characteristics of the service such as bandwidth, delay, availability and topological scope. Two types of SLSs are identified in MESCAL, extending previous work on intra-domain SLSs [4]:
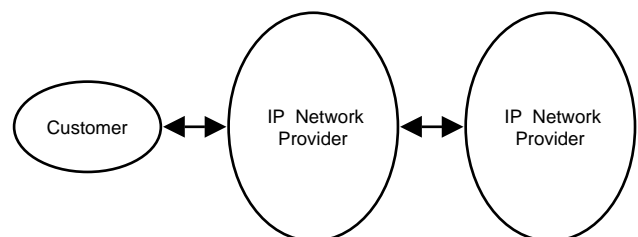

Fig. 1. MESCAL business model.
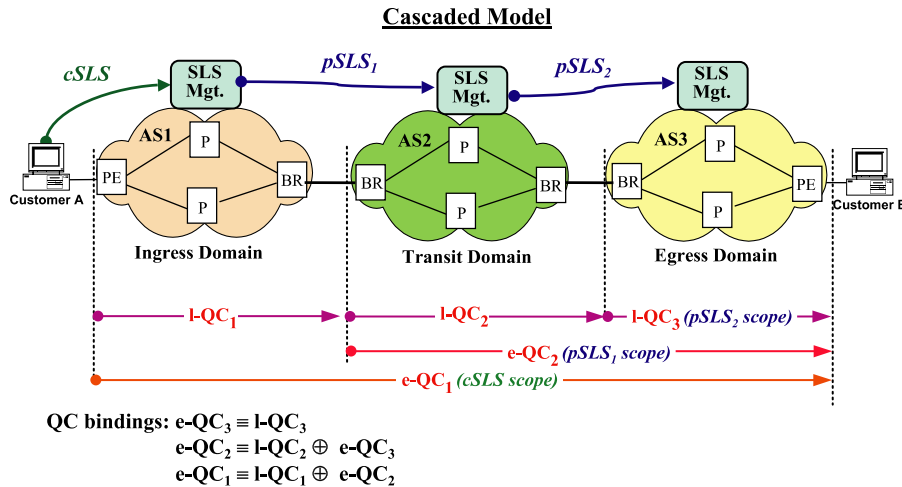
Cascaded Model

Fig. 2. Cascaded QoS peering model.

- Customer SLS (*cSLS*), established between end-customers and INPs; and
- Peer SLS (*pSLS*), established between INPs with the purpose of expanding the geographical span of their offered QoS services.

2.3. QoS classes

A *QoS class (QC)* denotes a basic network-wide *QoS transfer capability* of a single provider's domain. A QoS transfer capability is a set of attribute-value pairs, where the attributes express various packet transfer performance parameters such as one-way transit delay, packet loss and inter-packet delay variation (jitter), and their particular values. A provider domain's supported QCs are divided into *local QoS classes (l-QC)* and *extended QoS classes (e-QC)*, to allow us to capture the notion of QoS capabilities across domains:

- *l-QC* denotes a QoS transfer capability that is provided entirely within the local provider domain itself. The concept of l-QC could be compared to the Differentiated Services (DiffServ) Per Domain Behaviours (PDBs), and we shall see in Section 7 how l-QCs map onto the physical network implementation.
- *e-QC* denotes a QoS transfer capability that is provided by using both the local domain and other (service-peering) domains. An e-QC is provided by combining a local l-QC with appropriate l-QCs or e-QCs of other domains. The topological scope of an e-QC therefore usually extends outside the boundaries of the local domain.

2.4. The cascaded inter-domain QoS peering model

The term 'peering' is used throughout this paper to denote two providers interacting for the purpose of expanding the topological scope of their offered QoS-based services with financial settlements[1], i.e. peering here implies the existence of some form of customer–provider relationship [17]. There exist many models for the interconnection and service-layer interactions between providers for offering QoS services across domains. Conceptually, there exist a number of peering models such as hub, centralised, cascaded, and hybrid [18].

In general, providers prefer to offer services that reflect the current loosely coupled Internet structure. A *cascaded* model is therefore appropriate: this also has advantages of improved scalability and of allowing incremental deployment. Therefore, the MESCAL solution adopts a hop-by-hop cascaded model for the interactions between providers at both the service and network layers. In the cascaded model, each INP makes pSLS contracts with the immediately adjacent interconnected INPs. Thus, the QoS peering agreements are between adjacent neighbours, but not between providers more than 'one hop away'. This type of peering agreement is used to provide the QoS connectivity from a customer to reachable destinations that may be several domains away. Fig. 2 gives an overview of the operation of this model. l-QC₁, l-QC₂, and l-QC₃ are supported by AS1, AS2, and AS3, respectively. AS2 negotiates a contract (pSLS2) with AS3, enabling AS2's customers to reach destinations in AS3 with an offered e-QC₂. AS1 can then negotiate with AS2 (pSLS1), to enable AS1 customers to also reach destinations in AS3 with an e-QC₁, although at no point do AS1 and AS3 negotiate directly. Although not shown in the simple example of Fig. 2, in general there may be many options for combining a domain's local QoS capabilities (l-QCs) with those of adjacent providers. We use the term *QC Mapping* to mean the process of identifying this set of options. We then use

---

[1] This definition of peering is more generic than the one used today, which assumes peering between providers does not include any financial settlements, i.e. there is no customer-provider relationship.

the term *QC Binding* to describe the process of actually selecting which of the possible QC mappings are put into effect: for example, in the case of Fig. 2, associating AS2's internal l-QC$_2$ with the external l-QC$_3$ offered by AS3. QC binding might result in a number of QC bindings for a given e-QC, for example using different peers. This concept is illustrated later in Section 5.

3. The MESCAL functional architecture

This Section introduces from the perspective of a single provider an architecture that decomposes the functions required for the provision of inter-domain QoS services. This functional architecture (Fig. 3) is divided into three planes: management, control and data. The management plane includes offline functionality, typically located in management servers outside the network elements. Relevant functions are responsible for (a) interacting with customers and service peers to negotiate contracts and (b) implementing the business decisions of the INP through planning, dimensioning and configuring the network. The control plane covers intra- and inter-domain routing, handling the admission of traffic flows, and dynamic resource management including load distribution and capacity management functions. Typically, control plane functions are embedded within network equipment although they are not involved in packet-by-packet decisions. Finally, the data plane is responsible for per-packet treatment, and is configured by the control plane.

We now briefly describe the principal functional groupings within the architecture.

3.1. Service planning and QoS capabilities exchange

QoS-based Service Planning encompasses the business related activities responsible for defining the services a provider offers. These are specified according to the business objectives, and include l-QCs within the provider's own network, and e-QCs that result from combining its local QoS-based services with those offered by adjacent peers. Prior to any pSLS agreement with a peer, a provider has to *discover* from potential peer providers their QoS capabilities to various destination prefixes, and their associated costs. This is achieved using the *QoS Capabilities Discovery* function. Once l-QCs and e-QCs have been defined and engineered within a domain (by intra- and/or inter-domain TE), the *QoS Capabilities Advertisement* function is responsible for promoting the offered services so that customers and service peer providers are aware of the service offerings.

3.2. Network planning and provisioning

Network Planning includes the offline processes responsible for determining the type, quantity and geographical location of the physical resources (e.g. points of presence, IP routers and communications links) required by an INP. While many management activities can be achieved in an automated manner through network configuration, the implementation of planning decisions usually involves manual installation or configuration of physical equipment.

3.3. Offline traffic engineering

Traffic Forecast is responsible for aggregating and forecasting traffic demand. The set of subscribed cSLSs and pSLSs are retrieved from *SLS Order Handling*, and an aggregation process derives a traffic matrix between ingress and egress points of the domain. We define that part of the traffic matrix which lists flows that terminate in downstream ASs as an *external* traffic matrix (eTM). The traffic matrix is then used to calculate and provision the intra- and inter-domain resources needed to accommodate the traffic both from established SLSs and from those anticipated to be ordered during the provisioning cycle.

Traffic engineering is divided into inter- and intra-domain functions. Although we consider them as separate blocks, it is important to recognise that an optimal TE solution for end-to-end QoS requires the two to work together closely. For example, an inter-domain TE solution that assigns certain traffic flows to certain inter-domain links but results in some intra-domain links being overloaded is not a good solution.

Offline Inter-domain TE performs the QC mapping and QC binding operations described in Section 2.4 to construct potential e-QCs that meet the service requirements defined by *QoS-based Service Planning*. It then works with *Offline Intra-domain TE* to select a subset of these e-QCs while making optimal use of intra-domain and inter-domain network resources (this is QC binding). It also identifies a set of optimum pSLSs that need to be established with downstream providers.

Offline Intra-domain TE computes the intra-domain network configuration (routing constraints and capacity requirements per QC) that satisfies the predicted traffic demand.

3.4. Dynamic traffic engineering

Dynamic Inter-domain TE is responsible for inter-domain routing. An example of how this would be implemented is by a QoS-enhanced version of the BGP [15]. Dynamic inter-domain TE also dynamically performs load balancing between the multiple paths defined by *Offline Inter-domain TE*. It uses real-time monitoring information, changing appropriately the ratio of the traffic mapped to the inter-domain paths.

Dynamic Intra-domain TE is the dynamic management layer defined in [3]. It includes intra-domain routing, load balancing and dynamic bandwidth assignment for managing in real-time the resources allocated by *Offline Intra-domain*

TE, in order to react to statistical traffic fluctuations and special arising conditions.

3.5. SLS management

This includes two distinct phases: ordering, i.e. establishing contracts between peers, and invocation, i.e. committing resources before traffic can be admitted.

For ordering, *SLS Order Handling* implements the server side of the SLS negotiation process. Its purpose is to perform subscription level admission control. *SLS Order Handling* maps incoming SLS requests onto the e-QCs it can offer and investigates whether there is sufficient intra- and inter-domain capacity based on the resource availability matrix (RAM) received from TE for that e-QC. *pSLS Ordering* is the client side of the pSLS negotiation process: it receives requests from *Offline Inter-domain TE* for new pSLSs, and negotiates them with service peers, i.e. by communicating with *SLS Order Handling* in the peer domains.

Requests for invocation of pSLSs are handled by *pSLS Invocation*. Admission control is needed to ensure that the network is not overwhelmed with traffic; this allows the network to adopt a policy of overbooking resources at the subscription level. *SLS Invocation Handling*, the server side of the invocation process, contains the admission control algorithm, and receives requests from customers or peer providers for cSLS/pSLS invocations. It checks whether the invocation conforms to the subscribed SLS and if there is sufficient capacity in the local AS (and also on the inter-domain pSLS in the case of SLSs not terminated locally).

3.6. Data plane functions

Traffic Conditioning and QC Enforcement is responsible for packet classification, policing, traffic shaping and marking according to the conditions laid out in previously agreed SLSs and the invocation of those SLSs. At ingress routers *Traffic Conditioning* classifies incoming packets to their e-QC and marks them with the appropriate DiffServ Code Point (DSCP) for the required l-QC. At the egress router the *QC Enforcement* function may need to remark outgoing packets with the correct DSCP as agreed in the pSLS with the service peer. Thus, *QC Enforcement* implements the data-plane binding from l-QC to e-QC.

PHB Enforcement represents the queuing and scheduling mechanisms required to realise the different PHBs with the appropriate configuration.

3.7. Monitoring and SLAs assurance

Monitoring is responsible for node and network monitoring, collecting data at the request of other functional blocks and notifying them when thresholds are crossed on both elementary data and derived statistics. *SLS Assurance* compares the monitored performance and traffic statistics to the contracted QoS levels agreed in the SLSs to confirm that the network or service peer-networks are delivering the agreed service levels.

4. Solution options

The architecture described in Section 3 provides a framework for all the components required to implement inter-provider QoS, allowing coordination between neighbouring domains and the provision of end-to-end QoS through a cascaded model. The model is generic, allowing a variety of different performance guarantees. For example, residential customers may need to subscribe to QoS-based IP services in order to get to any reachable destination at any time simply with better-than-best-effort service levels. On the other hand, corporate customers may require hard upper bounds on QoS parameters and a constant bandwidth for supporting particular mission-critical services such as IP VPNs to a limited set of destinations. In order to satisfy a wide range of QoS requirements, and therefore potential customers, we have analysed three potential end-to-end approaches (Table 1), each of which is supported by a particular customisation of algorithms within the architecture proposed in Section 3. We call each such configuration a *solution option*. In this section, we describe the three solution options of Table 1.

4.1. Loose guarantees solution option

The first solution option relies on two concepts: these are *meta-QoS-classes* (m-QC) [19] and a QoS-enabled version of BGP, q-BGP (described in Section 6). The meta-QoS-Class denotes an abstract QoS-class, where the 'meta' qualification refers to the range of values of the QoS-class performance parameters (e.g. delay, loss, jitter). m-QCs describe in a quantitative or qualitative way ranges of QoS class performance parameter values, rather than particular values. Example of m-QCs are a stringent QoS class with delay value = 'very low' and loss value = 'very low', or a delay-sensitive QoS class with delay value = 'low' and loss value = 'any'.

As with all the solution options, the l-QCs are signalled within an AS by a dedicated DSCP value. In the loose guarantees solution option, these l-QCs are classified according to globally well-known m-QCs. Each provider advertises the m-QCs that it supports in its domains. Other providers can negotiate pSLSs with this domain in order to use its m-QCs. Once pSLSs have been agreed, the resulting QoS-enabled Internet can be viewed as a set of parallel planes each offering QoS service levels associated with a specific m-QC (Fig. 4) and running distinct instances of q-BGP. The pSLS
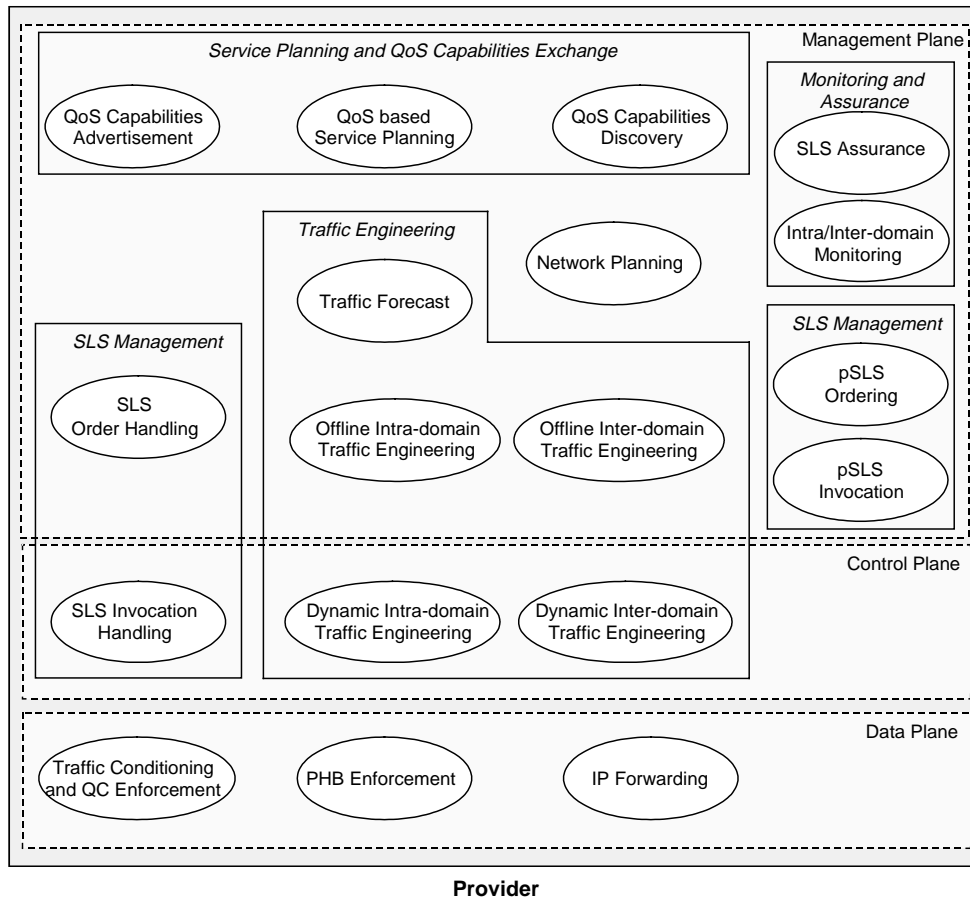
Fig. 3. Functional architecture for inter-domain QoS delivery.

agreement for an m-QC enables customers to send traffic to *all* destinations that are reachable with the given m-QC in the QoS-enabled Internet. pSLSs in the loose guarantees solution option are therefore valid for all

reachable destinations, not just a named subset. In this solution option, there is no need for a strict cascaded approach to build e-QCs. The function of Offline inter-domain TE in this solution option is to decide which pSLSs need to be agreed, and the amount of bandwidth (domain-to-domain, not end-to-end) they need. There are also no end-to-end QoS *guarantees* agreed: this is the fundamental difference between this option and statistical guarantee solution option described below in Section 4.2.

Table 1
MESCAL solution options

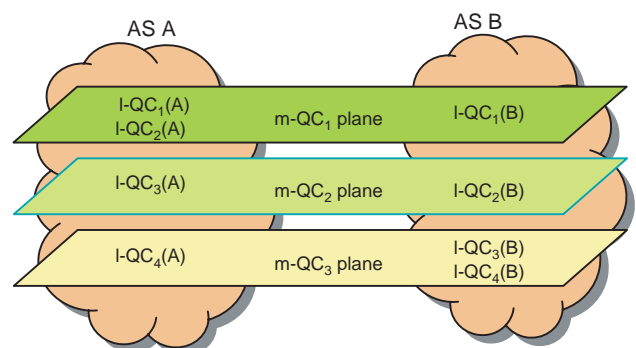| Solution option title	Usage	Characteristics		
		Topological scope	E2E QoS Performance	E2E Bandwidth
Loose guarantees	Improved Internet service for large user population	Any reachable destination	Qualitative	No guarantee
Statistical guarantees	Statistically bounded QoS for specified destinations	Specific destinations	Qualitative or Quantitative	Statistical guarantee
Hard guarantees	Hard guarantees based on paths/ tunnels	Specific destinations	Quantitative	Guaranteed

Fig. 4. m-QC planes in the loose guarantees solution option.

The core functions of the loose guarantees solution option are as follows:

- Establishing pSLSs, to enable the exchange of inter-domain connectivity information between service peer domains for each m-QC;
- Identification of the m-QC to which data packets and q-BGP announcements belong: this is achieved for data packets by means of the DSCP, and for q-BGP by means of an identifying field. The identifiers could either be globally known or only agreed between adjacent peers;
- Announcement of the network prefixes that can be reached within each m-QC plane: this is achieved through q-BGP, not through pSLS agreements;
- Data packet DSCP swapping at a domain's ingress and egress: at the ingress, the DSCP is swapped from a value that represents the m-QC to the l-QC DSCP value, and vice versa at the egress point.

When traversing a set of ASs, the QoS treatment experienced by an IP datagram is 'consistent' in all transit ASs. The term 'consistent' denotes that the treatment received by the IP packets in each AS conforms to the corresponding m-QC definition. By using the m-QC identifier included in the q-BGP UPDATE messages, each message can be processed within the context of the corresponding m-QC plane. QoS-related information can be inserted into q-BGP update messages, to facilitate the selection of the best possible end-to-end route.

4.2. Statistical guarantees solution option

The second approach is called the statistical guarantees solution option. It supports customers who have a higher QoS requirement than is provided by the loose guarantees solution option but a lower QoS requirement than the mission critical solution of Section 4.3. The QoS characteristics are therefore probabilistic rather than hard guaranteed. The statistical guarantees solution option assumes that each domain is engineered to support a number of l-QCs. End-to-end QCs are then built using the cascaded approach.

In order to be able to provide firmer guarantees than the loose guarantees solution option, this approach requires that pSLSs specify destination address prefixes: the QoS available to each prefix or group of prefixes can then be specified. The pSLS also includes the required QoS characteristics and the maximum bandwidth that the downstream AS offers to the specified destinations.

An INP that wishes to offer a particular e-QC to a specific set of destination prefixes may typically be able to bind any of a number of l-QCs with downstream e-QCs, provided that the combination of the chosen l-QC with the downstream e-QC meets the QoS requirements of the new e-QC that

the domain wishes to offer. We now illustrate this process of QC mapping and binding.

In Fig. 5, AS B and AS C (the downstream ASs) each offer their upstream neighbour AS A two e-QCs i.e. $eQC_1(B)$, $e-QC_2(B)$, and $e-QC_1(C)$, $e-QC_2(C)$, respectively to specific destination address prefixes. As a result of its Service Planning (Section 3.1), AS A decides to offer four eQCs to its customers or peer ASs. AS A discovers the QoS advertisements from ASs B and C using its QoS Capabilities Discovery function (Section 3.1), and its TE system decides to configure the following QC bindings:

(1) To support the offered $e-QC_1(A)$ by binding both its $l-QC_1(A)$ and $l-QC_2(A)$ with $e-QC_1(B)$: the two bindings are $l-QC_1(A) \oplus e-QC_1(B)$, and $l-QC_2(A) \oplus e-QC_1(B)$;
(2) To support both $e-QC_2(A)$ and $e-QC_3(A)$ by using one l-QC ($l-QC_3(A)$), which is bound to two downstream e-QCs, $e-QC_2(B)$ and $e-QC_1(C)$; and
(3) To implement $e-QC_4(A)$ by binding $l-QC_4(A)$ with $e-QC_2(C)$.

The advantage of using multiple bindings with different l-QC/e-QC combinations to produce one offered e-QC is to allow for both traffic engineering (e.g., load balancing) and resilience. In the above scenario, it is assumed that all the offered e-QCs from AS B and AS C support the destination address prefixes that AS A offers to its customers through its own defined e-QCs. In Fig. 5, when data packets that belong to $e-QC_2(A)$ and $e-QC_3(A)$ enter AS A, all of them are remarked with the *same* DSCP value that is mapped to $l-QC_3(A)$. When these packets leave AS A, the egress routers do not know how to differentiate between the two classes of packets that will be treated respectively with $e-QC_2(B)$ and $e-QC_1(C)$ offered by AS B and AS C respectively. To solve this QC splitting problem, we propose the use of spare DSCP values to be mapped onto the same physical l-QC ($l-QC_3(A)$).

Since this option offers statistical bandwidth guarantees only, it is expected that overbooking of resources will be allowed at subscription level; admission control is thus needed in order to ensure that the network is not overwhelmed with traffic. Offline TE is responsible for
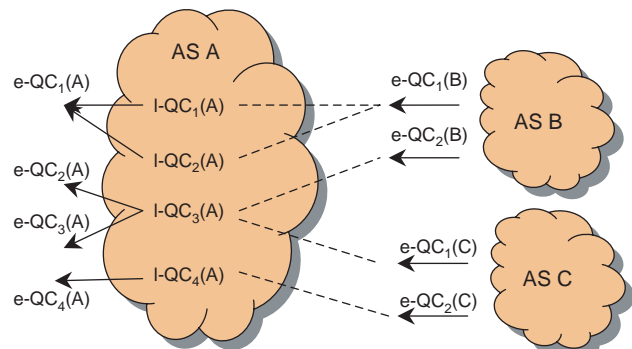

Fig. 5. QC mapping in the statistical guarantees solution option.

finding the optimum combination of l-QCs and downstream e-QCs in order to satisfy the requirements of the predicted traffic in each e-QC. This allows the domain to establish pSLSs with its peer domains and consequently optimally use the inter- and intra-domain resources.

Inter-domain routing is constrained by the pSLSs that are in existence at any time. The established pSLSs therefore influence inter-domain routing decisions, and in this option different paths to a given address prefix might be computed for different QCs. We anticipate that the solution would incorporate q-BGP in order to realise the offline TE decisions and provide dynamic operation.

4.3. Hard guarantees solution option

This solution option makes use of explicit and/or strict inter-domain MPLS LSP-TE tunnels [20]. The pSLS set-up in this solution option is the same as in the loose guarantees solution option as it provides more flexibility for LSP set-up. The open destination scope of pSLSs makes it possible to establish LSPs between any two points in the m-QC plane of the QoS-Internet as long as these points are within the coverage of that plane.

A Path Computation Element (PCE) is present in each domain and receives q-BGP announcements from all AS border routers (ASBRs) of the domain. Each domain receives the set of destinations that can be reached within each m-QC plane it supports, together with some aggregated QoS performance information. Thus, each PCE knows all destinations that can be reached within an m-QC plane together with their associated QoS performance characteristics. Each PCE communicates in the best-effort m-QC plane with its neighbours' PCEs using the proposed Path Computation Protocol (PCP) [21]. A full q-iBGP mesh between all ASBRs of a domain is set-up so that destinations learnt by a peering ASBR can be propagated to the other ASBRs in the same domain. QoS routes learned by q-BGP are made known to the IGP-TE in each domain so that a datagram can be routed to the correct egress point within a m-QC plane.

For setting-up an inter-domain LSP (between 'S' as the LSP head-end and 'D' as the LSP tail-end shown in Fig. 6), the PCE1 in the originating domain is asked to compute an inter-domain path in an m-QC plane meeting QoS requirements and optionally an associated bandwidth guarantee. PCE1 selects a possible path among the set of possible alternatives and identifies the ASBR1 as

the next-hop in the downstream domain. It then verifies that appropriate resources are available in its own domain and performs administrative pre-reservations in its domain. The LSP creation request is propagated to appropriate PCEs downstream. This is to communicate with the next hop PCE, requesting a path computation between its peering ASBR and the tail-end of inter-domain LSP (ASBR1 and 'D'). The request includes the AS's ASBR, the requester domain's PCE identifier [22], and the tail-end address of LSP. This procedure is repeated until the request reaches the destination PCE (PCE4). If a path satisfying the requested QoS requirements is found, each PCE returns the QoS path comprising of a list of LSRs to its upstream PCE. That is, the destination domain's PCE sends a reply message back to the upstream domain's PCE including the LSP loose path ('D', ASBR3) addresses in the message. It should be noted that at each domain, the intra-domain sub-path can also be added to the received path from the downstream domain's PCE and passed to the upstream domain's PCE. The next downstream domain's PCE does the same adding its own relevant ASBR addresses to the LSP loose path ('D', ASBR3, ASBR2). When the path information reaches the originating PCE, the LSP loose path ('D', ASBR3, ASBR2, ASBR1, 'S') is available and the originating PCE (PCE1) is in a position to request establishment of an inter-domain LSP using the above received path, i.e. either the LSP loose path or the whole computed path including intra-domain parts.

5. Simulation and evaluation of offline inter-domain traffic engineering

Having outlined the functional architecture and shown how solution options are the realisation of algorithms within the architecture, we now consider how to implement and optimise offline inter-domain traffic engineering. The approach described in this Section is optimised for the statistical guarantees solution option; it is also readily adaptable by simplification to the loose guarantees solution option.

5.1. Decomposition of offline inter-domain TE

Offline inter-domain traffic engineering optimises path selection while minimising overall cost. To achieve end-to-end QoS guarantees, it needs to work together with offline intra-domain traffic engineering to ensure that overall the TE solution is optimal. We now provide a decomposition of
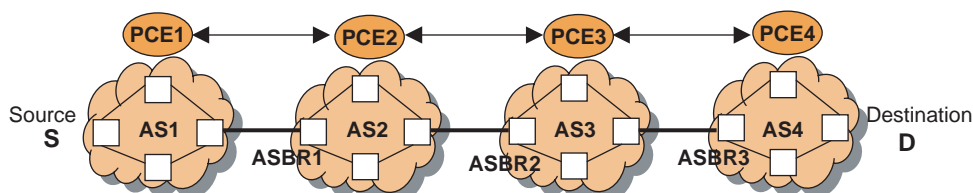

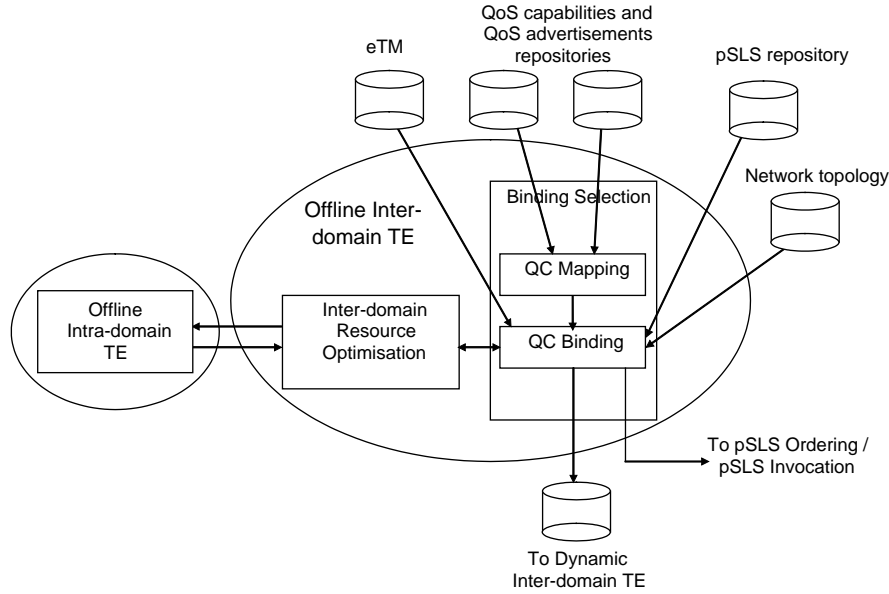Fig. 6. Hard guarantees solution option: LSP set-up through PCE-PCE communication.

Fig. 7. Decomposition of Offline Inter-domain TE.

the Offline Inter-domain TE functional block that was introduced in Section 3.3. We decompose the functionality into two main components: Binding Selection and Inter-domain Resource Optimisation (IDRO), shown in Fig. 7. As described in Section 2.4, Offline Inter-domain TE performs firstly QC Mapping and then QC Binding: these two functions together identify which l-QC and downstream e-QC combinations are used by each flow represented in the eTM (Section 3.3). In our decomposition, these two functions are described as Binding Selection. QC Mapping uses the information about the domain's own offered e-QCs and supported l-QCs (received from the QoS capabilities repository) and the information about downstream e-QCs offered by all adjacent domains (from the QoS advertisement repository). QC Binding also has as input the set of existing pSLSs, kept in a repository. The combined task of QC Mapping and QC Binding, performed by Binding Selection, is effectively the task of selecting an egress router and a pSLS for each traffic flow. In order to optimise the flow assignments, QC Binding employs IDRO; this is the algorithmic engine that computes a set of paths and returns its cost to QC Binding. For IDRO, we propose a genetic algorithm for the statistical guarantees solution option, and later in this section, we consider its performance, comparing it with two reference algorithms: random assignment, and a brute force algorithm.

While inter-domain TE optimises inter-domain resources (QC bindings and peer pSLSs), it is necessary to ensure that (a) there are sufficient intra-domain resources (l-QC capacity) between the anticipated ingress and selected egress routers, and (b) that the intra-domain configuration to meet the selected inter-domain bindings is not sub-optimal. An iterative algorithm therefore runs between Offline Inter- and Intra-domain TE (Fig. 3).

The output from Offline Inter-domain TE, used as input to Dynamic Inter-domain TE, is used to configure the physical network described later. QC Binding can also identify whether any new pSLSs need to be ordered during the current provisioning cycle, and sends any such requests to pSLS Ordering; this function is not however considered further in this paper.

In the remainder of this Section, we present algorithms for Offline Inter-domain TE, and the results for an implementation that considers as QoS metrics both delay and bandwidth.

5.2. Traffic engineering algorithms

5.2.1. QC mapping
Fig. 8 illustrates an example set of binding candidates identified by QC Mapping, where the only QoS parameter considered is the additive function, delay. A combination of l-QC with downstream e-QC that satisfies an offered e-QC is only considered as a binding candidate if the downstream e-QC appears in one or

Fig. 8. QC Mapping example.

more QoS advertisements. The IDRO algorithms described below further only consider a binding candidate if there are one or more potential pSLSs that support the binding candidate's downstream e-QC and the pSLS supports one or more destination prefixes listed in the eTM.

5.2.2. Cost functions

We consider here a total of three cost functions: the first is related to inter-domain costs, the second reflects the intra-domain cost, and the third cost function minimises inter-domain link utilisation.

In apportioning a multi-domain QoS constraint between two or more domains a number of approaches are possible. Pongpaibool and Kim [13] defined three policies that a domain can implement:

- A *least-effort policy*, in which a domain selects the lowest intra-domain QoS class (l-QC in our terminology) it supports that is compatible with the desired inter-domain QoS (offered e-QC in our terminology). A downstream domain is thereby forced to employ a higher QoS class (downstream e-QC) in order to meet the overall QoS constraint;
- A *most-effort policy*, in which a domain selects the highest intra-domain QoS grade (l-QC) compatible with the desired inter-domain QoS (offered e-QC). A downstream domain is thus able to employ a lower QoS class (downstream e-QC) to meet the overall QoS constraint;
- An *equal distribution policy*, where the responsibility for delivering the required level of QoS is split evenly between domains.

It is our view that in any commercial scenario, such as the current multi-provider Internet, an INP will seek to minimise its own effort, and therefore cost, in providing any given level of QoS. We therefore consider pricing mechanisms that reflect the QoS burden incurred by a downstream domain that carries traffic of a high QoS class (downstream e-QC) on towards its destination. This requires an explicit cost associated with each pSLS and a cost associated with the intra-domain traffic. We therefore extend the above policy definitions by using 'low effort policy' to mean one in which a domain selects a low l-QC or QoS grade and therefore a high downstream e-QC, and a 'high effort policy' to mean one in which a domain selects a high l-QC or QoS grade and a low downstream e-QC. Thus in general, depending on the relative costs of intra-domain provision and inter-domain pSLSs, a given individual flow in the eTM might be assigned an l-QC/e-QC binding on the basis either of a low effort policy or alternatively a high effort policy, in such a way that the *overall* cost of carrying all traffic flows defined in the eTM is minimised. For example, it might make economic sense for an INP to employ a high effort policy on at least some of its flows if the total cost of using a high QoS l-QC with a low cost downstream e-QC is less than the cost of using a low cost l-QC with a high cost downstream e-QC. Our approach therefore generalises that of [13], allowing a domain to choose the effort policy or policies that it finds the most advantageous.

In line with the above discussion our first cost function considers the cost incurred by neighbouring domains of providing a given level of QoS. In this paper we present results for one example pSLS cost model. For a single pSLS, the cost is given by $\sum_{t \in S}$ where $t(i,k,e)$ is the bandwidth of a traffic flow from ingress i to a destination prefix k with an e-QC e, C_s is the cost per unit bandwidth of pSLS s and S is the set of flows assigned to pSLS s. The total pSLS cost Ω summed across all pSLSs is therefore

$$\Omega = \sum_s \left(C_s \sum_{t \in S} t(i,k,e) \right). \tag{1}$$

Our second cost function complements the first and represents the cost of using intra-domain resources. We use a simple illustrative model to reflect the higher cost of low-delay l-QCs:

$$\Phi = K \sum_{\text{flows}} \frac{t(i,k,e)}{\delta(t)} \tag{2}$$

where $t(i,k,e)$ is the bandwidth taken by the flow and $\delta(t)$ is the l-QC delay assigned to flow t. K is a scaling factor used to give an approximately even weighting between intra- and inter-domain costs.

We can now observe that if we sum a pSLS cost and an intra-domain cost, we obtain a cost function that reflects both the inter-domain and intra-domain costs, and which therefore allows an effort policy to be implemented in an optimal way. Our approach is thereby one of minimising the total cost, represented by the sum of these two cost functions.

Finally, our third cost function considers explicitly the objective of minimising the overall inter-domain link utilisations. This reduces maximum queuing delays and allows for statistical fluctuation in traffic beyond that forecast in the traffic matrix. For the link utilisation cost a number of standard models exist, and we have used one based on [7]. In our work, we truncate the continuous piecewise linear cost at a maximum utilisation of 1.0, and instead mark any traffic engineering solution that exceeds the inter-domain link bandwidth as 'invalid'. We also penalise high utilisations (> 0.9) more heavily than [7]. The cost for a single link is given by $\theta(x)$ where x is the utilisation and $\theta(0) = 0$, $\theta(0.3) = 0.3$, $\theta(0.7) = 1.3$, $\theta(0.9) = 4$, $\theta(1.0) = 20.0$, being linearly interpolated within these points (Fig. 9). The inter-domain link utilisation cost function is then
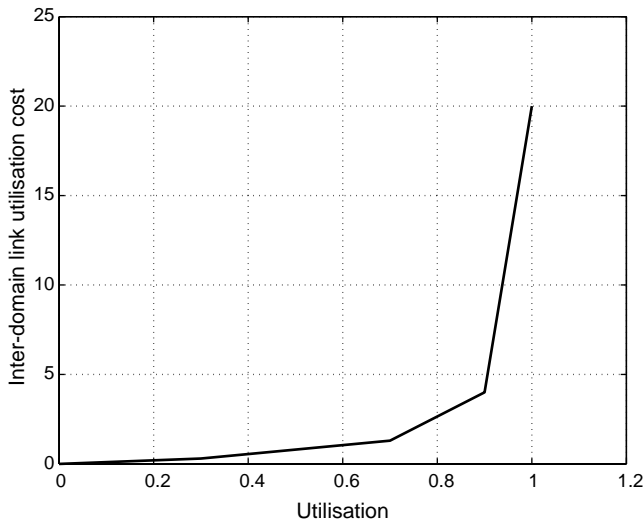
Fig. 9. Inter-domain link utilisation cost function.

$$\Theta = \sum_j \theta(x_j) \tag{3}$$

where x_j is the utilisation on egress link j.

5.2.3. IDRO Algorithms

The overall objective of IDRO is to ensure that flows described in the eTM are optimally assigned to the inter-domain resources (pSLSs on inter-domain links). To achieve this it takes each flow recorded in the eTM (from a given ingress point to a given destination prefix requiring a given bandwidth or data rate with a given e-QC) and assigns it to an l-QC (intra-domain behaviour) and downstream e-QC (offered by a downstream domain) on a particular egress link, whilst ensuring that the inter-domain link bandwidth constraint is met on each inter-domain link and the pSLS bandwidth constraint is met for each pSLS. IDRO then calculates the configuration cost using the functions as described above in Section 5.2.2, calling Offline Intra-domain TE.

5.2.3.1. IDRO: optimisation problem formulation.
The optimisation problem may be stated as follows: given an eTM containing a set of flows each from an ingress id i to a destination prefix k requiring bandwidth or data rate $t(i,k,e)$ with e-QC e, assign each flow to an l-QC l (intra-domain behaviour) and a downstream e-QC o on some egress link j, so as to minimise the cost function while meeting pSLS, intra-domain link utilisation and inter-domain link utilisation constraints. The combination of downstream e-QC o and egress link j is equivalently represented by a pSLS id $s_{j,o}$. Specifically, we minimise the sum of our cost functions subject to the constraints:

$$\sum_{r \in s_{j,o}} t(i,k,e) \le B_s \tag{4}$$

where B_s is the maximum capacity of the pSLS $s_{j,o}$, that is, the pSLS on egress link j that provides downstream e-QC o, and for inter-domain links

$$\sum_{s_{j,o} \in s_j} \sum_{t \in s_{j,o}} t(i,k,e) \le B_j \tag{5}$$

where B_j is the capacity of the inter-domain link j.

This problem is in effect a generalised GAP problem when generalised cost functions are used: this has been proved to be NP-hard [23], and an approach such as a genetic algorithm is required to solve it.

5.2.3.2. Genetic algorithm.
Genetic algorithms provide a heuristic mechanism for solving complex optimisation problems [24–26]. Each potential solution to a problem is represented by a set of values known as a chromosome; for example, in our case the chromosome consists of the assignment of each aggregate traffic flow in an eTM to a particular l-QC, egress router and a pSLS. The chromosome is composed of individual genes; in our case a gene would be the assignment of a single aggregate traffic flow to its l-QC, egress router and pSLS. The genetic algorithm heuristic comprises the following. An initial population of N randomly generated chromosomes is generated. Each of these solutions is then used as the configuration of the system under investigation (in our case, the inter-domain configuration), and a fitness function is calculated that quantifies the 'goodness' of the solution represented by the chromosome. Once the fitness function has been calculated for all N chromosomes a new generation of chromosomes is produced, as follows.

The chromosome population is divided into three sections: the best, the middle, and the worst. The best and middle chromosomes are passed unchanged to the next generation. The poorest chromosomes (i.e., those with the worst fitness function) are discarded. Processes of crossover and mutation (described below) are applied to the best and middle chromosomes to generate new chromosomes for the next generation. This process results in a new population of N chromosomes, and the process of generating a new generation is repeated until convergence. The chromosome with the best fitness function is then the best (or fittest) chromosome, and represents the best obtained configuration of the system under investigation.

In the crossover process, two chromosomes are randomly selected, one from the best section of the population, and one from the middle section. Genes are randomly selected from each of the chromosomes to generate a new chromosome. The probability of selecting from the fitter parent chromosome (i.e. from the 'best' section) is called the crossover probability, p_c. In mutation, genes are randomly changed, with some mutation probability p_m. The pseudo-code for the algorithm is given in Fig. 10.

```
Generate initial population of N random chromosomes
While (not converged or Nmax iterations) {
    Calculate fitness (cost) for each chromosome
    Sort chromosomes by fitness value
    Perform reproduction
    – Split chromosomes into 3 groups: best, middle, worst
    – Best and middle propagate unchanged to next generation
    For each worst chromosome {
    – Discard old chromosome
    – Generate new chromosome as follows:
        • Randomly select 2 parents, one from best and one from
          middle groups
        • For each row in the eTM {
            • Crossover: randomly select individual genes (eTM flows
              with l-QC & pSLS assigned) from either parent; or
            • Mutation: with probability pm, generate a random new
              gene (eTM flow) instead of performing crossover
            }
        }
    }
}
```

Fig. 10. Genetic algorithm pseudo-code.

The effectiveness and convergence rate of the genetic algorithm depends on the values of N, p_c and p_m. Previous research suggests typical satisfactory values to be $150 \le N \le 300$, $0.5 \le p_c \le 0.8$, and $0.001 \le p_m \le 0.1$ [26]. In our results we have used $N = 250$, $p_c = 0.6$ and $p_m = 0.01$.

5.2.3.3. Random assignment algorithm. The random assignment algorithm is used as a reference algorithm for comparison purposes. The pseudo-code for the random assignment algorithm is given in Fig. 11.

5.2.3.4. Brute force assignment algorithm. Pseudo-code for the brute force assignment algorithm is given in Fig. 12. The run time for this algorithm scales exponentially with the number of flows in the eTM, and it has therefore only been used to validate the other algorithms at low scale. No results are presented in this paper for this algorithm.

5.3. Evaluation of TE algorithms

5.3.1. Simulation scenario

The network topology for the simulations is shown in Fig. 13, focusing on the inter-domain connectivity. We assume a moderate sized AS with 20 adjacent ASs. The AS

```
While Nmax iterations not exceeded {
While not all flows assigned {
    Randomly select a eTM flow
While not all binding candidates considered {
    Assign this flow to a randomly chosen binding
    candidate that meets the eTM's e-QC
    – i.e. assign a l-QC and e-QC
While not all pSLSs considered {
    Randomly select a pSLS that:
    (1) supports the e-QC of the binding candidate;
    (2) matches the destination prefix;
    (3) has spare bw capacity in the pSLS; and
    (4) has spare bw capacity on the inter-domain link
}}}}
```

Fig. 11. Random assignment algorithm pseudo-code.

```
While not all bc and pSLS combinations considered {
For each row in the eTM {
    Assign the flow to first / next binding candidate
    – i.e. assign a l-QC and e-QC
    Assign first / next pSLS that supports the e-QC and
    matches the destination prefix
    }

    Calculate validity (i.e. whether entire solution meets pSLS
    and inter-domain link capacity constraints) and cost of this
    solution
    }
    Select lowest cost solution
```

Fig. 12. Brute force assignment algorithm pseudo-code.

under test supports two l-QCs (25 and 50 ms delays), and as a result of its service planning (Section 3.1) wishes to offer two e-QCs for its inter-domain flows (100 and 175 ms). Each adjacent AS is connected to the AS under test by either 1 or 2 links, giving a total of 27 inter-domain links, each of whose link bandwidth is set in the range 150–300 units. A number of destination prefixes are reachable through each adjacent AS (there may be other ASs en-route to the final destination prefix, but these are not relevant to our model). Each AS is able to reach between 30 and 60 of the prefixes. This reflects the observation that a small number of destination prefixes are responsible for a large fraction of an AS' outbound traffic volume [10]. Although in reality the destination prefixes will in general overlap each other, for simplicity here we assume they are disjoint.

Each adjacent AS is assumed to support a subset of three downstream e-QC delays. For simplicity, the set of supported delays is identical in each adjacent AS, being set to any of 50, 75 and 125 ms (Fig. 8). QoS advertisements for each link are generated based on a random combination of downstream e-QCs and random pSLS costs; for the QoS advertisements announced by any individual adjacent AS, the cost of a higher QoS class (i.e. lower delay) is set higher than the cost of a lower QoS class. Each pSLS has a bandwidth in the range 0 to 300, and the pSLS cost is set to a value between 1 and 10 per unit bandwidth. This results in overbooked pSLSs that support a total bandwidth that is 1.9 times the inter-domain link capacity. In the evaluation described here, each QoS advertisement is assumed to have
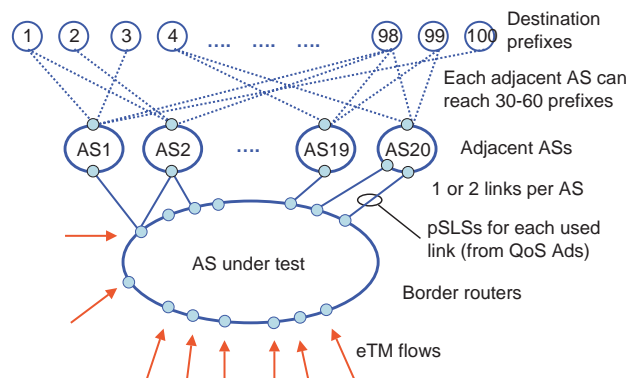

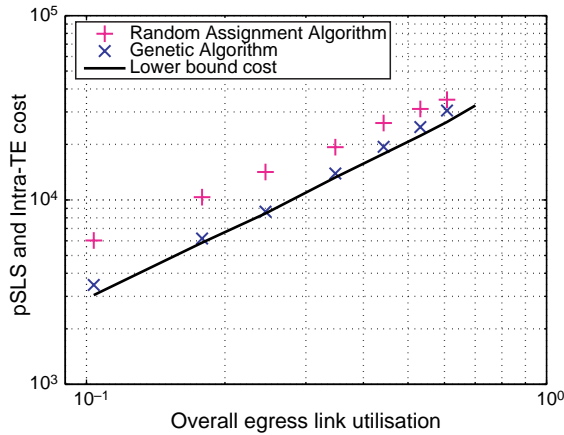Fig. 13. Simulation network topology.

Fig. 14. Offline Inter-domain TE validation.

Fig. 15. pSLS cost and Intra-domain TE cost.

resulted in the establishment of a pSLS, resulting in a total of 47 pSLSs being available to the 20 adjacent ASs. Finally the entire system is driven by a set of eTM flows randomly generated in such a way that the destination prefix in each eTM entry can be reached through one or more pSLSs supported by at least one adjacent AS. Each flow requires either a 100 or 175 ms e-QC to one of the 100 remote destination prefixes, and has a bandwidth requirement randomly selected in the range 1–40.

The behaviour of the algorithm can be validated by considering a simplified set of QCs, in which only a single downstream e-QC is employed. We assume that the intra-domain links have sufficient capacity to carry all flows. We simplify the problem by assuming that all destination addresses in the eTM can be carried by the set of lowest cost pSLSs that have a total bandwidth equal to the total bandwidth in the eTM. Finally we relax the problem constraints by allowing a single eTM flow to be partially assigned to more than one pSLS. The result of these simplifications is to decouple the task of assigning an l-QC to a given eTM flow from the task of pSLS selection, and an analytically solvable approximation to the problem can be produced. In this case, all flows are optimally carried within the AS using the cheapest l-QC (i.e. the one with the highest delay), and for the inter-domain link the flows are all assigned to the set of lowest cost pSLSs. By considering only the two cost functions pSLS cost and intra-domain TE, we can calculate using a spreadsheet a lower bound cost, shown in Fig. 14 by the solid line. This lower bound cost is better than the brute force solution. The motivation for calculating a lower bound is to observe how close the genetic algorithm approaches this simplified approximation. We see that the genetic algorithm produces results close to this lower bound.

5.3.2. Results

We now present results for the test scenario of Fig. 13 with the full set of QoS classes shown in Fig. 8. We first consider only two cost functions: pSLS cost Ω
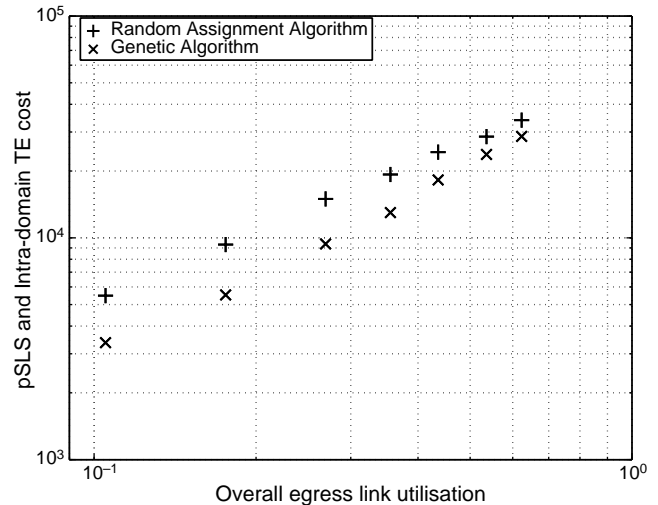
and intra-domain TE cost Φ. Fig. 15 shows how the sum of these costs varies as the total eTM traffic increases. The x-axis is normalised by dividing the total eTM flow by the sum of the capacities of the inter-domain links. The genetic algorithm has a lower cost than the random assignment algorithm at all values of utilisation. We note in passing that the brute force algorithm is only computationally feasible at very low utilisation, and that at this point, the genetic algorithm solution successfully matches the cost of the brute force solution.

In essence, the genetic algorithm identifies solutions where a flow can be assigned to a low-cost combination of l-QC and downstream e-QC. A destination prefix is in general reachable with a given downstream e-QC through a number of different pSLSs, and each of these pSLSs is offered by an adjacent AS at one of a number of different pSLS costs. The genetic algorithm identifies the pSLS with the lowest cost.

We can observe this behaviour by analysing the utilisation of each pSLS. In Fig. 16 the 47 pSLSs are shown, arranged in ascending order of cost per unit bandwidth. For each pSLS, the assigned bandwidth is shown for the random assignment algorithm and for the genetic algorithm. We see that the random assignment algorithm has distributed the flows over all pSLSs approximately evenly. However, the genetic algorithm has weighted the flows towards the lower cost pSLSs. In fact, the random assignment algorithm has assigned only 18% of the traffic to the pSLSs with cost per unit bandwidth of 2.2 or less, whereas the genetic algorithm has assigned 85% of the traffic to these pSLSs.

However, the flow assignments are made without consideration of the maximum inter-domain link utilisation and have resulted in the genetic algorithm assigning flows such that some links are heavily utilised (Fig. 17). This can be corrected by introducing the third cost function Θ so that the total cost function is the sum of the pSLS cost, Intra-domain TE cost, and inter-domain link utilisation (Fig. 18). The link
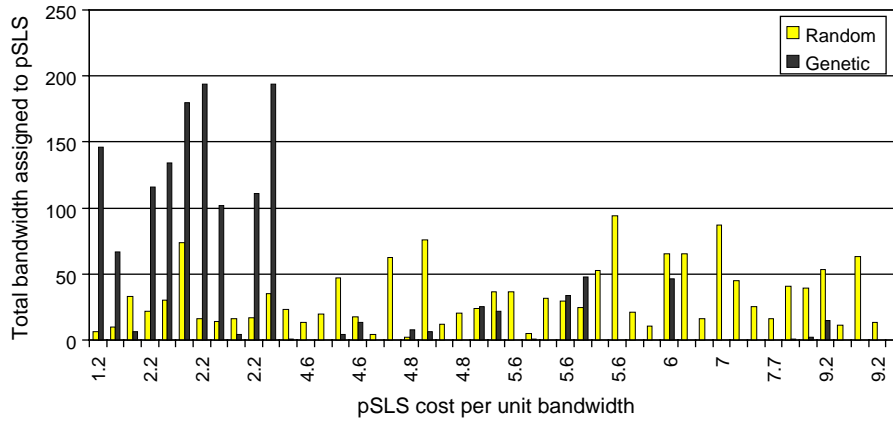
Fig. 16. Comparison of pSLS utilisation in random and genetic algorithms.

utilisation cost function is scaled so that all three components are given approximately equal weight. By introducing the link utilisation function, the peak link utilisations are reduced (Fig. 19), with the worst link utilisation from the genetic algorithm reduced from 99% to 69%.

In summary, we have shown that in a simplified scenario the genetic algorithm obtains results that are close to the lower bound solution; that in a more complex scenario the GA can be used to obtain offline QoS-aware traffic engineering solutions that are of significantly lower cost than a random approach; and that we can reduce the maximum inter-domain link utilisation by representing this utilisation in the cost function, minimising the total of the inter-domain pSLS costs, intra-domain TE costs and inter-domain link utilisation costs.

6. QoS-inferred border gateway protocol (q-BGP)

A number of authors have proposed approaches for enhancing BGP [27] capabilities to support new QoS requirements such as those discussed in this paper. We now propose a set of enhancements to BGP, which we call QoS-inferred BGP (q-BGP). This set satisfies the generic requirements that q-BGP should be dynamic, scalable, able to propagate topology changes without any significant impact on the existing best-effort protocol, and applied to any kind of inter-domain QoS delivery solution that is based on an exchange of QoS-related information between domains. The QoS-related information exchange can occur either at the service level during the pSLS negotiation phase or at the routing level using q-BGP. The place this exchange occurs depends on the inter-domain QoS information delivery mechanism deployed. We therefore identify two groups of QoS delivery mechanisms. The first group (group-1) only requires propagation of an identifier specifying an m-QC plane that was agreed earlier during the pSLS negotiation phase: QoS performance characteristics will have been negotiated in the pSLS but are not exchanged in the routing level. The second group (group-2) requires propagation both of QoS performance characteristics and its
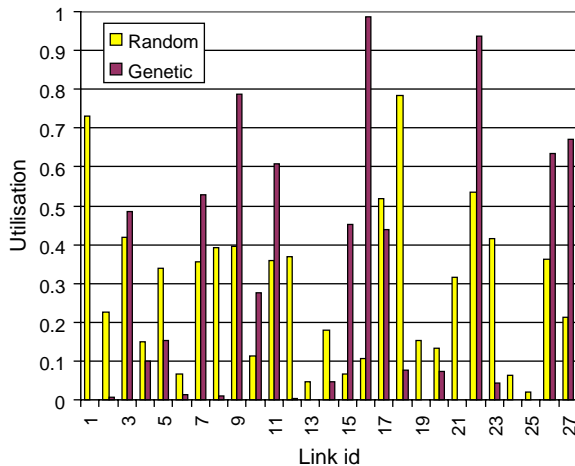

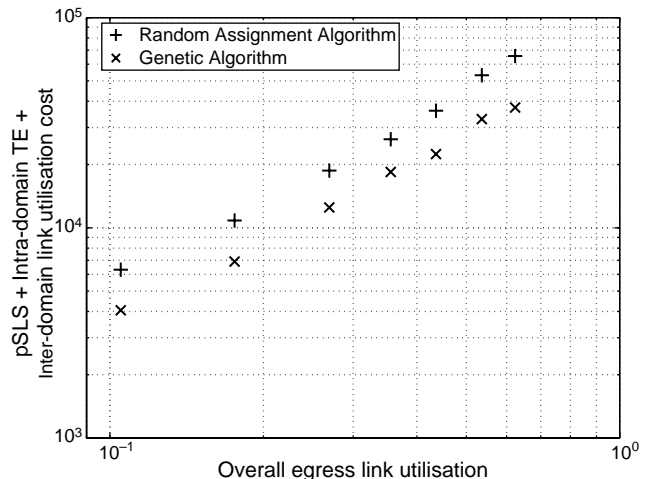Fig. 17. Link utilisations (pSLS cost and intra-TE cost only).

Fig. 18. pSLS cost, intra-domain TE, and inter-domain link utilisation costs.
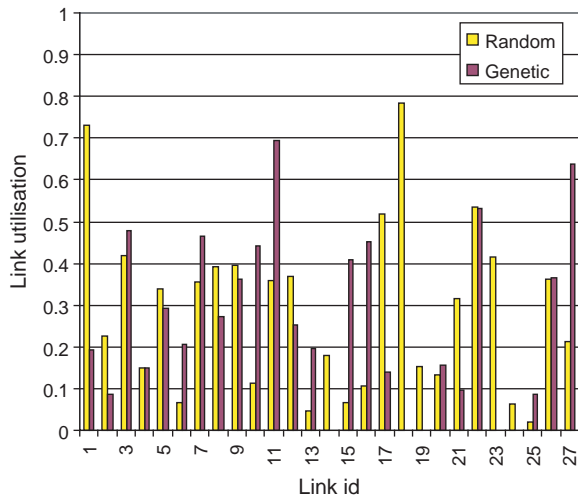
Fig. 19. Link utilisations (including inter-domain link utilisation in cost function).

associated identifier. In group-2 the type of the QoS-related information to be exchanged is agreed during the pSLS negotiation phase, but not the actual values.

Therefore, q-BGP provides the following features:

- *QoS service capabilities*: since peering entities need to know about each other's QoS service capabilities, q-BGP identifies the solution option(s) that a domain provides, and indicates what information can potentially be carried by the q-BGP messages;
- *QoS class (QC) identifier*: this is used to distinguish the e-QCs (or m-QC planes) that have been bought by/from service peers;
- *QoS performance characteristics*: these are a set of QoS characteristics values, such as one-way packet loss, delay and inter-packet delay variation.

In order to implement the above features, we adopted as a starting point the new optional transitive attribute QoS_NLRI described in [15]. However, the proposal has been extended as follows:

- Ref. [15] allows only one QoS performance characteristic to be sent per q-BGP announcement, whereas it is more general to support a list of multiple QoS performance characteristics in one single q-BGP UPDATE message;
- Unlike the proposal in [15], the q-BGP specification allows propagation of pre-negotiated e-QC information between service peers. PDB, m-QC and/or e-QC identifiers can thus be announced by q-BGP;
- The proposal described in [15] adopts multiple paths [28]. This is not the case in the current q-BGP specification;
- A new field called "QoS information length" is added, to control the list of QoS performance characteristics that are included in a q-BGP UPDATE message.

q-BGP can advertise QoS information either statically, set by administrators, or dynamically, obtained through measurement. It is our current view that dynamic exchange of QoS information via q-BGP is not recommended because this could generate route oscillation and thereby impact the stability of routing table. Providers should instead configure q-BGP advertised QoS information statically, with information being valid over the medium term.

Since QoS information is now exchanged in q-BGP UPDATE messages, the route selection process has to be modified in order to take it into account. The q-BGP route selection process differs according to whether the group-1 or group-2 solution is used. In the case of group-1, the q-BGP route selection process is very similar to the classical route selection process currently used for BGP, i.e the q-BGP route selection process will choose the route that minimises the AS_PATH hops for each QC plane. In group-2, since several QoS parameters may be advertised for a given destination, the process examines each QoS parameter in a prioritised order. Thus, the route selection process chooses the best routes by examining initially the highest priority QoS parameter. If several routes have the same weight for the highest priority parameter, the second priority parameter is considered, and this process is repeated as necessary until a route is selected.

7. Dynamic inter-domain traffic engineering: testbed experimentation

MESCAL has defined a number of functional entities, protocols and algorithms [29] that support and implement the functional architecture of Section 3. A number of these entities have been implemented and evaluated in a testbed and simulators. The testbed is used to:

- Validate q-BGP, including its capacity to advertise and select inter-domain QoS routes;
- Validate the meta-QoS-class concept;
- Validate the implementation of l-QCs in each domain, using Linux traffic control features [30];
- Validate, at the data plane level, DSCP swapping (marking/remarking) between ASs in order to signal an inter-domain QC (e-QC or m-QC).

7.1. Implementing and evaluating dynamic inter-domain TE

7.1.1. Network topology

The MESCAL testbed comprises eight autonomous systems (ASs) shown in Fig. 20. The testbed is designed to represent the hierarchical Internet topology: thus AS4 and AS6 represent Tier-1 providers and have more network resources allocated. AS1 represents a Tier-3 provider. The remaining ASs are Tier-2 providers. A number of Linux-based routers are used in the testbed. Each router has
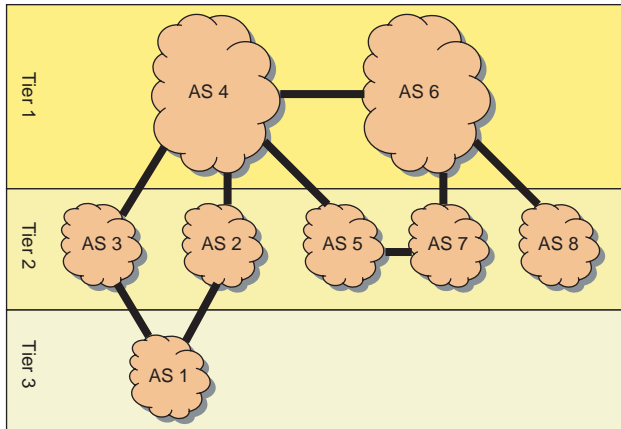
Fig. 20. MESCAL testbed.

a number of interfaces and represents a single AS, except AS4, which is composed of three routers. In a single router AS, the router acts as the ASBR of its domain. In AS4, no QoS-inferred intra-domain routing protocol is used. All testbed routers are additionally connected through a dedicated interface to an administrative network that is used for testbed configuration and maintenance. q-BGP is activated at the boundaries of each domain and a full q-BGP mesh is configured within the AS4 domain.

In addition, traffic generators are connected to each router via a separate dedicated interface, and are used for emulating customer traffic and generating background load.

7.1.2. QoS Configuration

All routers have DiffServ capabilities for traffic classification, traffic conditioning and various scheduling disciplines. To implement different classes of service, the HTB (Hierarchical Token Bucket) scheduling discipline [31] is used rather than CBQ (Class Based Queuing) [32]. Preliminary tests showed that the Linux CBQ implementation had difficulty handling more than 1.5 Mbit/s of IP traffic. In addition, the Linux implementation of TBF (Token Bucket Filter) has a 1 Mbit/s handling limitation. Ingress re-marking is achieved by using IPFILTER [33], and DSCP egress re-marking is achieved by using DSMARK queuing discipline [34]. Four l-QCs are configured in each AS. A bandwidth threshold is also configured for each l-QC. Shaping and policing are also configured respectively at the egress and ingress interfaces in order to control bandwidth usage as agreed during pSLS negotiation between two adjacent ASs.

7.1.3. Routing configuration

q-BGP is used as the inter-domain routing protocol. Each AS has at least one q-eBGP session (i.e. q-BGP session between two different routers of different ASs) with its adjacent ASs. q-iBGP sessions (i.e. q-BGP session between two routers of the AS) are established between all AS4 border routers. Static routes are configured in AS4 instead of

an IGP. The q-BGP configuration is pSLS driven. Thus, the q-BGP configuration file is retrieved from agreed pSLSs stored in the pSLS repository. q-BGP can advertise QoS routes either with dynamic QoS values (obtained through monitoring) or static ones (QoS values configured by an administrator having a specified period of validity).

7.2. q-BGP implementation

One of the main objectives of the testbed is to check the validity of the modified BGP protocol (q-BGP) and its capacity to advertise and select inter-domain QoS routes as explained in the following Sections.

A commercial routing protocol suite (ZebOS) [35] has been used to implement q-BGP features, principally:

- QoS service capabilities;
- QoS_NLRI messages;
- l-QC definition;
- Instantiation: instructs q-BGP to instantiate a RIB for a particular QC plane as defined by the domain's administrator;
- Route selection process: the classical route selection process is modified in order to take into account the QoS parameters. The q-BGP implementation allows a 'precision,' x, to be configured in the route selection process. This precision is used when comparing two routes having similar QoS characteristics: if the two routes have QoS values that are within x% of each other they are said to be identical;
- CLI command: new CLI commands have been implemented in order to ease q-BGP configuration.

7.2.1. q-BGP configuration

In order to activate q-BGP between two peers, the following parameters must be configured in border routers:

- A list of local QCs: in the control plane, each l-QC is defined by a DSCP value and a list of QoS performance characteristics (e.g. any or all of minimum one-way delay (owd), maximum owd, average owd, and inter-packet variation delay). Any QoS performance characteristic can be declared to be optional or mandatory. A priority value is also associated with each QoS parameter. Fig. 21 is a configuration example of an l-QC identified by a DSCP value of 16, and characterised by a bandwidth (reserved rate) of 2000 that is set as the highest priority QoS parameter. Thus, in this example,

```
ZebOS# configure terminal
ZebOS(config)# local-qos-class 5
ZebOS(local-qos-class)# qos-class-id 16
ZebOS(local-qos-class)# qos-inf reserved-rate 2000 priority 1 mandatory precision 5
ZebOS(local-qos-class)# qos-inf max-owd 50 priority 2 optional
```

Fig. 21. l-QC configuration.

ZebOS# configure terminal

ZebOS(config)# route-map QOS_IN permit 3

ZebOS(config-route-map)# match qos-class-id QCID 5

ZebOS(config-route-map)# set local-qos-class 4

Fig. 22. Route-map example.

reserved rate is the first selection criterion considered during q-BGP route selection process. This parameter is also configured as mandatory, i.e. q-BGP announcements injected in this l-QC plane must specify a value for reserved rate parameter, or else the q-BGP announcement will be dropped. A maximum one-way delay (optional parameter) is also configured with a priority less than the reserved rate. Since this parameter is optional, a q-BGP announcement will be injected in this l-QC plane even if the announcement does not indicate a one-way delay value.

- A list of IN/OUT QoS ROUTE-MAPs: these describe the QC Bindings determined by Offline Inter-domain TE by mapping them to DSCP values. The inter-domain DSCP values are agreed between adjacent ASs during the pSLS negotiation phase and are used in q-BGP announcements (the QoS class identifier). Thus a dedicated DSCP value signals a given e-QC or m-QC on the inter-domain links. At the ingress interface q-BGP must therefore be aware of the mapping between agreed inter-domain DSCP values (representing an e-QC) and the local DSCP value (representing the corresponding l-QC). Similarly at the egress, q-BGP must be aware of the mapping between the local DSCP value (l-QC) and the outgoing DSCP value (the downstream e-QC of the neighbouring AS). Therefore, we have modified the classical BGP ROUTE-MAP (as implemented by ZebOS) in order to activate the DSCP binding operations described above. Two sets of ROUTE-MAP rules are configured per DSCP binding, one each at the ingress and egress. Fig. 22 illustrates an ingress ROUTE-MAP rule, setting the QoS class identifier value (contained in

a q-BGP QoS_NLRI attribute) to 4 if the QoS class identifier field of received q-BGP announcements is set to 5. This set of configuration statements instructs q-BGP to inject the announcements received with a QoS class identifier of 5 into the q-RIB identified by the local-qos-class 4.

- List of network prefixes and QoS planes in which the prefixes will be advertised.

7.2.2. Experimental results

We now describe a set of tests that validate the functional behaviour of q-BGP. DSCP swapping, QoS aggregation and route selection process have been tested, and conform to q-BGP specifications. In addition to the best effort plane, q-BGP also advertises QoS routes in appropriate QC planes. q-BGP selects the best route per QC plane per destination. Fig. 23 shows the output of the 'show ip bgp' command, and illustrates QoS aggregation and identifies routes that have been selected by q-BGP for l-QC 41 (defined in AS4). Selected routes are identified by the '>' symbol and the Next Hop column identifies the IP address of the corresponding next-hop router.

Fig. 23 shows an example of a q-RIB. Our q-BGP RIB implementation contains more information than the classical the BGP RIB. The following columns are particularly noteworthy:

- QC-id: this indicates a DSCP value. That means that a given route will be used for routing datagrams towards destinations identified by the IP address in the "Network" column only if the datagram contains the correct DSCP value. For example, the routing information for the 1.1.1.0/30 prefix will only used if the DSCP value in the IP header is set to 34.
- A list of QoS parameter values, e.g avg-owd, max-owd and jitter: these values indicate the aggregated values of the QoS parameters that will be experienced if this path is selected to route IP datagrams. The symbol '–'

```
MESCAL43# sh ip bgp local-qos-class 41
BGP table version is 0, local router ID is 43.43.43.1
Status codes: s suppressed, d damped, h history, p stale, * valid, > best, i - internal
Origin codes: i - IGP, e - EGP, ? - incomplete

local-qos-class 41 (loose solution options) :
   Network        QC-id Next Hop      Metric LocPrf Weight avg-owd max-owd jitter   loss  available reserved min-owd Path
                                                                                  -rate    -rate     -rate

*>i1.1.1.0/30     34 2.2.2.1                100      0     355     475     21      448     2000      3000      51    2 1 ?
*>i1.1.1.4/30     34 2.2.2.1          0      100      0     330     440     16      349     2500      3000      36    2 ?
*>i2.2.2.0/30     34 2.2.2.1          0      100      0     330     440     16      349     2500      3000      36    2 ?
*>i3.3.3.0/30     34 41.41.41.5       0      100      0     300     400     10      150     4000      6000      16    ?
*   5.5.5.0/30    34 43.43.43.2                       0     310     470     20      448      -        3500      -     6 7 ?
*>i                34 42.42.42.6       0      100      0     305     434     15      299      -         -        -     5 ?
*  i6.6.6.0/30    34 41.41.41.2       0      100      0     305     434     15      299     3000      4200      -     6 ?
*>                 34 43.43.43.2       0             0     305     434     15      299     3000      4200      -     6 ?
*  i6.6.6.4/30    34 41.41.41.2       0      100      0     305     434     15      299     3000      4200      -     6 ?
*>                 34 43.43.43.2       0             0     305     434     15      299     3000      4200      -     6 ?
```

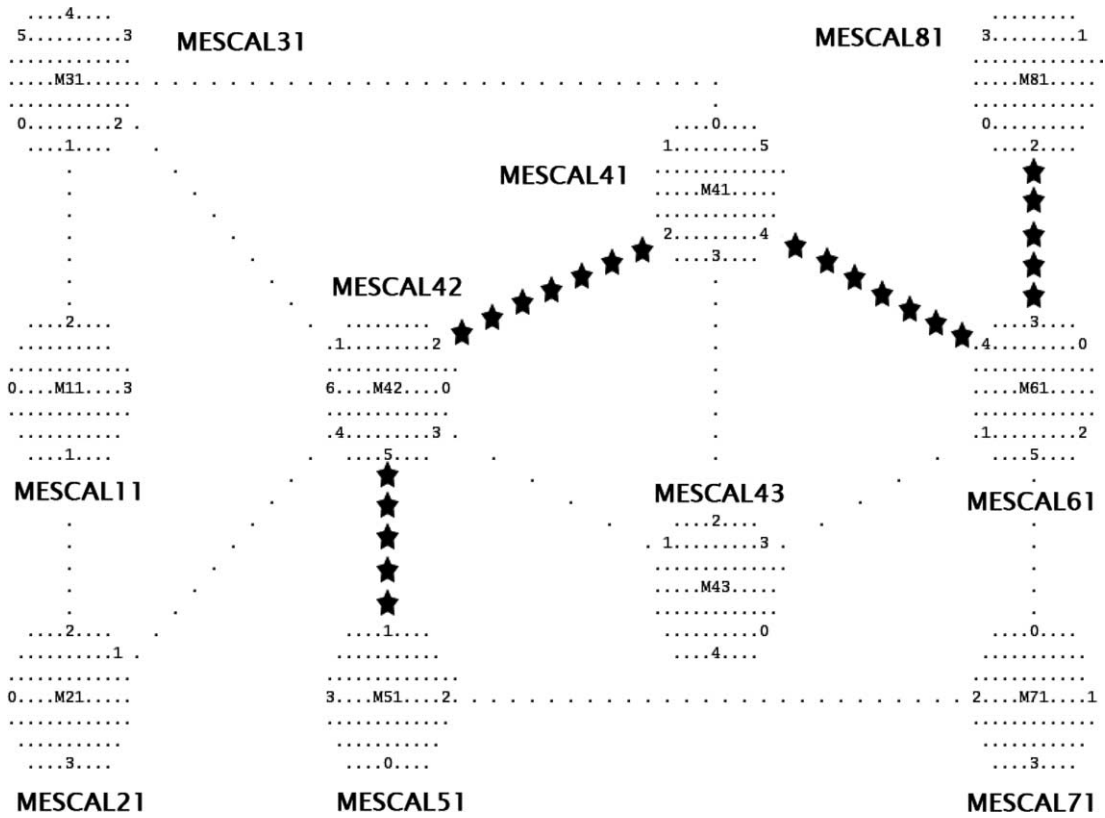Fig. 23. q-BGP Routing Information Base (q-RIB) of a border router of AS4.

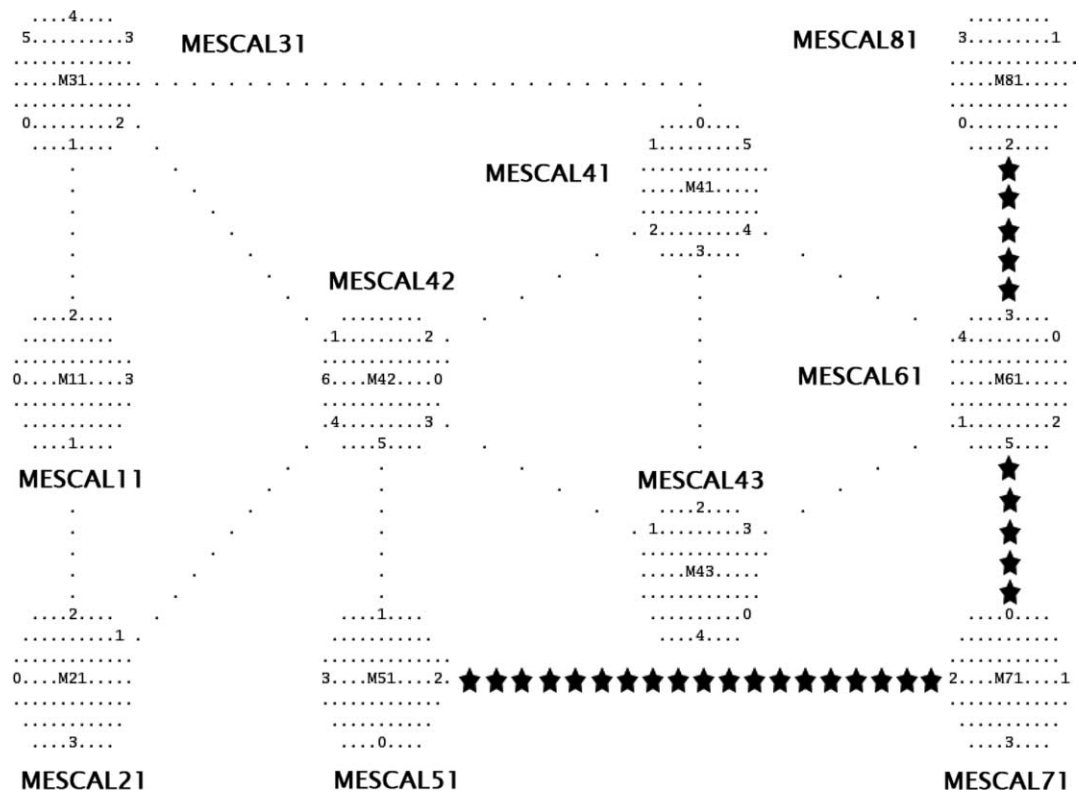Fig. 24. Ping request in the best effort plane.

Fig. 25. Ping request with DSCP value of 10.

indicate that at least one AS in the path to this destination does not support this QoS parameter.

We now present results from a ping tracker, developed in MESCAL, that tracks sent and received ICMP ECHO requests. This illustrates the path used by a ping request in order to reach given destinations. In the presented results, q-BGP has been used to select routes. Figs. 24 and 25 show the routers in the testbed; the first digit in the router name identifies the AS number, thus Mescal61 is the router that comprises AS6.

Fig. 24 is a snapshot of the ping tracker showing the path of a best effort (DSCP value of 0) ping request sent from AS8 towards a destination prefix in AS5. The path of the ping request is indicated by the "*" symbols. This figure shows that q-BGP selected the path AS8-AS6-AS4-AS5, using the classical route selection process. We compare this path with that achieved by q-BGP for a QoS-enabled ping. For this, all routers in the testbed are configured to prioritise the average one-way delay (avg-owd) parameter. The router in AS6 (MESCAL61) received q-BGP advertisements that gave an avg-owd to AS5 of 200 ms via AS4 and 150 ms via AS7. Hence, when a QoS ping request (with a DSCP value of 10) is sent from AS8, q-BGP selects the path AS8-AS6-AS7-AS5, because this path minimises the value of the avg-owd parameter. (Fig. 25).

8. Conclusions

In this paper we have defined a vocabulary for inter-domain QoS provisioning, and used it to present an architecture that describes the key functions required to provide inter-domain QoS across the multi-provider commercial Internet. We have described three approaches, compatible with the architecture, which provide various levels of QoS guarantees: these were (1) a set of parallel QoS planes where each plane conforms to a well-known understanding, summarised as a meta-QoS class; (2) a solution that provides statistical guarantees to given destination prefixes; and (3) a solution that uses MPLS TE tunnels to provide hard guarantees to given individual destinations. We then formulated an offline inter-domain traffic engineering problem, primarily optimised for the statistical guarantees solution option, and showed how a genetic (evolutionary) approach could be used to perform QoS-aware inter-domain TE, balancing the costs incurred by a provider between the costs within their own network (the intra-domain TE costs) and costs external to their network (represented by pSLS costs). Finally, we have shown using a testbed how QoS-inferred BGP can be implemented across multiple domains to support QoS delivery.

Acknowledgements

References

[1] G. Huston, Next Steps for the IP QoS architecture, IETF Informational RFC 2990, November 2000.

[2] R. Atkinson, S. Floyd, IAB concerns & recommendations regarding Internet research & evolution, IETF Internet-Draft draft-iab-research-funding-01.txt, July 2003.

[3] P. Trimintzios, et al., A management and control architecture for providing IP differentiated services in MPLS-based Networks, IEEE Communications Magazine 39 (2001) 80–88.

[4] D. Goderis, et al., Service level specifications semantics, parameters, and negotiation requirements, IETF Internet-Draft, draft-tequila-sls-02.txt, February 2002.

[5] M. Kodialam, T.V. Lakshman, Minimum interference routing with applications to MPLS traffic engineering, Proc. INFOCOM 2000, Israel 2000.

[6] A. Elwalid, C. Jin, S. Low, I. Widjaja, MATE: MPLS Adaptive Traffic Engineering, Proc. INFOCOM 2001 2001.

[7] B. Fortz, M. Thorup, Optimizing OSPF/IS-IS weights in a changing world, IEEE Journal on Selected Areas in Communications 20 (2002) 756–767.

[8] Z. Wang, et al., Internet traffic engineering without full mesh overlaying, Proc. INFOCOM, Alaska 2001.

[9] B. Quoitin, C. Pelsser, L. Swinnen, O. Bonaventure, S. Uhlig, Interdomain traffic engineering with BGP, IEEE Communications Magazine 41 (2003) 122–128.

[10] N. Feamster, J. Borkenhagen, J. Rexford, Guidelines for interdomain traffic engineering, ACM Computer Communications Review, Oct 2003.

[11] P. Fuzesi, K. Nemeth, N. Borg, R. Holmberg, I. Cselenyi, Provisioning of QoS enabled inter-domain services, Computer Communications 26 (2003) 1070–1082.

[12] K-H. Ho, N. Wang, P. Trimintzios, G. Pavlou, M. Howarth, On egress router selection for inter-domain traffic with bandwidth guarantees, Proc. High Performance Switching and Routing Conference, IEEE, Phoenix Arizona 2004.

[13] P. Pongpaibool, H.S. Kim, Providing end-to-end service level agreements across multiple ISP networks, Computer Networks 46 (2004) 3–18.

[14] L. Xiao, K-S. Lui, J. Wang, K. Nahrstedt, QoS extension to BGP, IEEE Proc ICNP2002 Conference 2002.

[15] G. Cristallo, C. Jacquenet, The BGP QoS_NLRI attribute, IETF Internet-Draft, draft-jacquenet-bgp-qos-00.txt, February 2004.

[16] E. Mykoniati, et al., Admission control for providing QoS in IP DiffServ networks: the TEQUILA Approach, IEEE Communications Magazine 41 (2003) 38–44.

[17] G. Huston, Interconnection, peering and settlements, Internet Protocol Journal 2 (1999) 2–16.

[18] H. Asgari et al., Issues in MESCAL inter-domain QoS delivery, technologies, inter-operability and financial settlements, MESCAL project deliverable D1.4, available at www.mescal.org, 2004.

[19] P. Levis, M. Boucadair, P. Morand, P. Trimintzios, The Meta-QoS-Class concept: a step towards global QoS inter-domain services, Proc. IEEE International Conference on Software, Telecommunications and Computer Networks SoftCOM 2004.

[20] M. Boucadair, P. Morand, A Solution for providing inter-AS MPLS-based QoS tunnels, IETF Internet-Draft, draft-boucadair-pce-interas-00.txt, October 2004.

[21] M. Boucadair, P. Morand, Inter-AS PCE Communication Protocol, IETF Internet-Draft, draft-boucadair-pcp-interas-00.txt, October 2004.

[22] M. Boucadair, P. Morand, Path Computation Service discovery via Border Gateway Protocol, IETF Internet-Draft, draft-boucadair-pce-discovery-00.txt, October 2004.

[23] T.C. Bressoud, R. Rastogi, M.A. Smith, Optimal configuration for BGP route selection, Proc. IEEE INFOCOM 2003.

[24] P.C. Chu, J.E. Beasley, A genetic algorithm for the generalised assignment problem, Computers Operations Research 24 (1997) 17–23.

[25] M. Ericsson, M.G.C. Resende, P.M. Pardalos, A genetic algorithm for the weight setting problem in OSPF routing, Journal of Combinatorial Optimization 6 (2002) 299–333.

[26] X.-H. Lin, Y.-K. Kwok, V.K.N. Lau, A genetic algorithm based approach to route selection and capacity flow assignment, Computer Communications 26 (2003) 961–974.

[27] Y. Rekhter, T. Li, A Border Gateway Protocol 4 (BGP-4), IETF RFC 1771, March 1995.

[28] D. Walton, et al., Advertisement of multiple paths in BGP, IETF Internet-Draft draft-walton-bgp-add-paths-01.txt, Work in Progress, November 2002.

[29] M. Howarth, et al., Initial specification of protocols and algorithms for inter-domain SLS management and traffic engineering for QoS-based IP service delivery and their test requirements, MESCAL project deliverable D1.2, available at www.mescal.org, 2004.

[30] B. Hubert, G. Maxwell, R. van Mook, M. van Oosterhout, P.B. Schroeder, J. Spaans, Linux Advanced Routing & Traffic Control HOWTO, Published v0.9.0, March 2002. Available at http://lartc.org/.

[31] M. Devera, HTB User Guide, May 2002, available at http://luxik.cdi.cz/~devik/qos/htb/userg.pdf.

[32] S. Floyd, V. Jacobson, Link-sharing and resource management models for packet networks, IEEE/ACM Transactions on Networking 3 (1995) 365–386.

[33] IP Filter, v4.1.3. Documentation available at: http://coombs.anu.edu.au/~avalon/.

[34] W. Almesberger, J. Hadi Salim, A. Kuznetsov, Differentiated Services on Linux, IETF Internet Draft, draft-almesberger-wajhak-diffserv-linux-01.txt, June1999.

[35] ZebOS Advanced Routing Suite, www.ipinfusion.com.