Available online at www.sciencedirect.com
ScienceDirect
Computer Communications 30 (2007) 3757–3777
computer communications
www.elsevier.com/locate/comcom

Inter-autonomous system provisioning for end-to-end bandwidth guarantees ☆

Kin-Hon Ho *, Michael Howarth, Ning Wang, George Pavlou, Stylianos Georgoulas

Centre for Communication Systems Research, University of Surrey, Guildford, Surrey GU2 7XH, United Kingdom

Received 14 September 2006; received in revised form 31 August 2007; accepted 5 September 2007
Available online 20 September 2007

Abstract

This paper addresses the issue of provisioning end-to-end bandwidth guarantees across multiple Autonomous Systems (ASes). We first review a cascaded model for negotiating and establishing service level agreements for end-to-end bandwidth guarantees between ASes. We then present a network dimensioning system that uses traffic engineering mechanisms for the provisioning of end-to-end bandwidth guarantees. The network dimensioning system solves two problems: (1) the economic problem of how to determine the optimum amount of bandwidth that needs to be purchased from adjacent downstream ASes at a minimum total cost; (2) given the available bandwidth resources within and beyond the AS as a result of (1), the engineering problem of how to assign bandwidth guaranteed routes to the predicted traffic while optimizing the network resource utilization. We formulate both as integer-programming problems and prove them to be NP-hard. An efficient genetic algorithm and an efficient greedy-penalty heuristic are, respectively, used to solve the two problems and we show that these perform significantly better than simple heuristic and random approaches.

Keywords: End-to-end bandwidth guarantees; Bandwidth provisioning; Traffic engineering

1. Introduction

Today's Internet is a collection of Autonomous Systems (ASes), providing a best-effort service to traffic that is typically sent across multiple ASes. Each AS is an administrative region governed by its own policies and has full control of its own resources. The emerging future-generation Internet is expected to provide end-to-end Quality of Service (QoS) guarantees. In situations where stringent end-to-end QoS is required, ensuring that an adequate bandwidth is guaranteed by each AS along the inter-AS route is essential to achieve relevant performance targets [1]. Yet in practice, an AS is only capable of provisioning bandwidth guarantees within its own network. Extending bandwidth guarantees beyond its boundary requires the AS to agree the supply of sufficient bandwidth from downstream ASes. This bandwidth supply is likely to have a financial cost and therefore there is an economic incentive for an AS to carefully select its downstream ASes so as to minimize the cost of using that bandwidth.

Having purchased access to sufficient bandwidth from downstream ASes, the AS needs to utilize both this purchased bandwidth and its own network capacity in the most effective way in order to provide bandwidth guarantees for the predicted traffic. Internet Network Providers (INPs) thus need to optimize the utilization of these resources. Traffic Engineering (TE) is an effective technique to optimize IP operational network performance and subsequently improve network QoS capabilities [2]. INPs can thus use TE as an effective means for bandwidth guarantee provisioning while optimizing network resource utilization.

☆ The contents of this paper have been partially published in the IEEE Workshop on High Performance Switching and Routing (HPSR) 2004 [43] and the IEEE Workshop on IP Operations and Management (IPOM) 2004 [44].

* Corresponding author. Tel.: +44 1483 683 641; fax: +44 1483 686 011.
E-mail addresses: K.Ho@surrey.ac.uk (K.-H. Ho), M.Howarth@surrey.ac.uk (M. Howarth), N.Wang@surrey.ac.uk (N. Wang), G.Pavlou@surrey.ac.uk (G. Pavlou), S.Georgoulas@surrey.ac.uk (S. Georgoulas).

0140-3664/$ - see front matter Crown copyright © 2007 Published by Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2007.09.008

Concatenation of bandwidth guarantees between ASes makes possible to provide an end-to-end guarantee between a source and a destination. These guarantees across ASes owned by different INPs require some level of agreement between ASes, usually summarized in a negotiated Service Level Agreement (SLA). An SLA is an agreement between a user and a provider that describes the characteristics of a service, specifying in particular the supported QoS and the associated cost. However, given that the Internet consists of thousands of ASes, SLA negotiation between ASes has to be carefully managed in an effective and scalable manner. In this paper, we adopt a cascaded negotiation model which allows ASes to build up end-to-end SLAs that provide end-to-end bandwidth guarantees. In this model, apart from route reachability information, each AS receives from adjacent downstream ASes a set of what we call bandwidth offers to named remote AS destinations. If an AS decides to accept a bandwidth offer, an SLA is established between the two ASes. The AS can then in turn make bandwidth offers to its upstream (customer) ASes; these offers reflect both the AS' own resources and the SLAs established with the downstream ASes. The full set of SLAs enables all the ASes to support traffic with end-to-end bandwidth guarantees. However, the AS' tasks of making appropriate decisions on which bandwidth offers to accept, how much bandwidth to purchase and how to allocate the bandwidth among traffic aggregates are non-trivial. Inappropriate bandwidth offer selection or traffic assignment could result in, respectively, high cost or poor resource utilization. In order to obtain the best solutions, we propose a network dimensioning system that incorporates optimization modules that solve the two following problems:

- how to determine an appropriate amount of bandwidth to be purchased from each bandwidth offer so that the total cost of the bandwidth is minimized;
- given the knowledge of the available intra-AS bandwidth and the bandwidth purchased from downstream ASes, how to assign routes to the predicted traffic aggregates so that bandwidth demand is met while optimizing network resource utilization.

We call these two problems the *Inter-AS Bandwidth Provisioning* and *Traffic Assignment* problems, respectively. Our proposed network dimensioning system enables ASes to move from trial-and-error to a systematic approach for provisioning their end-to-end bandwidth guarantees.

The main contributions of this paper can be summarized as follows:

- We propose a systematic network dimensioning system that can be used by ASes to achieve effective provisioning of end-to-end bandwidth guarantees. The network dimensioning system formulates two problems that, respectively, provide economic and engineering optimization, namely the inter-AS bandwidth provisioning and traffic assignment problems.
- We show that a heuristic approach can be used to solve the inter-AS bandwidth provisioning problem. To illustrate this, we use a genetic algorithm. Our proposed genetic algorithm optimizes the bandwidth provisioning with 5–30% and 75–90% less cost than a conventional heuristic and a random-based algorithm, respectively.
- We use a greedy-penalty heuristic algorithm to solve the traffic assignment problem. The proposed greedy-penalty heuristic results in 10% less total bandwidth consumption than a random-based algorithm.

The rest of the paper is organized as follows. In Section 2, we review related work. In Section 3 we review a scalable cascaded model for the negotiation and agreement of inter-AS bandwidth guarantees. We then proceed to present the decomposition of the proposed network dimensioning system in Section 4. In Sections 5 and 6, we formulate the inter-AS bandwidth provisioning and traffic assignment problems, and propose algorithms to solve them. We present our simulation study to evaluate the proposed algorithms in Section 7. Finally, we conclude the paper in Section 8.

2. Related work

We classify related work into three areas: QoS provisioning through traffic engineering, inter-AS bandwidth provisioning and inter-AS traffic engineering.

2.1. QoS provisioning through traffic engineering

A significant amount of work has been conducted on QoS provisioning through TE. The European Union IST TEQUILA project [3] presented an architecture for QoS provisioning in Differentiated Services (DiffServ) networks, with an integrated approach that brought together service management and traffic engineering to achieve intra-AS QoS provisioning [4]. Similar intra-AS QoS provisioning through TE has also been applied to IP-based production networks [5]. In [6], the authors proposed an automated Multi-Protocol Label Switching (MPLS)-based TE manager, TEAM, in DiffServ networks. A set of algorithms was proposed to provide QoS and to achieve better resource utilization. Their design and implementation was evaluated in an IP QoS testbed. In [7], the authors demonstrated the benefits of traffic engineering. They evaluated traffic engineering in the presence of QoS-based routing schemes compared with the standard destination-based routing. The authors observed that TE can provide 20–50% network capacity savings.

The above works, however, have mainly focused on QoS provisioning within an AS. Less work has been performed on end-to-end QoS provisioning across multiple ASes, although this is an important issue because most Internet traffic traverses a number of immediate ASes before reaching its destination [8]. The EU IST MESCAL project [9] presented a functional architecture for inter-AS QoS provisioning [10] through TE. This is based on a cascaded model for service negotiation between ASes. In [11], the authors proposed a scalable Differentiated Services architecture for end-to-end QoS provisioning. The architecture uses an agent, called a Bandwidth Broker [12], that employs a TE mechanism to optimize resource allocation within an AS. QoS provisioning between two ASes relies on inter-AS signaling, admission control and resource reservation. Both two works proposed a framework to enable end-to-end QoS provisioning. However, important network planning issues, such as how much bandwidth is required to support the traffic demand and how to assign traffic to the existing bandwidth, were not addressed.

2.2. Inter-AS bandwidth provisioning

Inter-AS bandwidth provisioning is one of the key issues addressed in this paper. Bandwidth provisioning is the process by which an AS determines the amount of bandwidth needed to support traffic with desired level of performance targets [13]. Most prior work has addressed intra-AS bandwidth provisioning, with only few proposals at the inter-AS level. Duan et al. studied a bandwidth provisioning problem for service overlay networks, taking into account various factors such as SLAs, QoS and traffic demand distributions [14]. Given the stochastic traffic demand and the pre-determined routes between any source and destination gateways, they determined the appropriate amount of bandwidth to be purchased from the underlying network ASes to provide end-to-end value-added QoS across multiple ASes. Although the appropriate amount of bandwidth is purchased, their work did not address how to optimally assign traffic to the purchased bandwidth in order to achieve better network utilization.

2.3. Inter-AS traffic engineering

The TE problem in this paper can be viewed as the optimization of traffic exiting an AS [8,15]. This optimization problem has attracted attention in recent years. Bressoud et al. addressed offline traffic engineering with the objective of optimizing inter-AS transit traffic flow [16]. The requirement was to determine an optimal selection of outgoing links and border routers to be used for egress of transit traffic where the selection minimizes provider network utilization and balances the load of traffic flows exiting the service provider across the selected egress links while respecting capacity constraints. Akella et al. showed that selecting the appropriate set of Internet Service Providers for traffic routing yields a performance improvement [17,18]. Goldenberg et al. [19] proposed novel smart offline and online routing algorithms to optimize cost and performance for multihomed ASes. Feamster et al. proposed an inter-AS outbound traffic optimization model for network engineering [20]. Most of these proposals target at engineering best-effort type traffic. However, they did not take QoS guarantees as business objective, which we will consider in this paper.

2.4. Inter-AS resource reservation

An appropriate amount of resources may be reserved in order to provide QoS guarantees for a traffic flow. However, as the number of traffic flows across ASes increases tremendously, resource reservation must be performed in a scalable way. BGRP is a protocol proposed for scalable inter-AS resource reservation [48]. The fundamental concept of BGRP is that it builds a sink tree for each of the leaf ASes. Each sink tree aggregates bandwidth reservations from all data sources in the network. BGRP is scalable since each router needs to maintain only the sink tree information rather than the states for millions of traffic flows. However, BGRP focuses only on reducing the amount of state information that routers are required to maintain but ignores other important issues during resource reservation. As a result, some enhancements or new proposals have been made to improve these deficiencies. An extension to the BGRP, called BGRPP [49], was proposed to further reduce the resource reservation signaling overheads and consequently limits the protocol's bandwidth utilization as well as the induced processing load. In addition, the authors in [50] proposed a bandwidth over-reservation approach to reduce the signaling load associated with establishing and maintaining reservations.

2.5. Contribution of this paper

Our work differs from the above papers in that we consider end-to-end bandwidth guarantees and we propose an integrated approach to bring together inter-AS bandwidth provisioning and TE (both intra- and inter-AS TE) as an effective means to provision the guarantees across multiple ASes. This results in new TE optimization problems and solutions. By developing TE techniques for provisioning end-to-end bandwidth guarantees, our work advances the state of the art in this area.

3. Cascaded negotiation model

The provision of end-to-end bandwidth guarantees requires each intermediate AS on the path from the source AS to the destination AS to guarantee the agreed bandwidth. However, this cannot be realized without first negotiating and agreeing SLAs among the ASes. Since the Internet is a collection of a large number of ASes, attention needs to be paid to how to manage such negotiation and SLA establishment in an effective and scalable manner. In this paper, we adopt a cascaded model, as proposed by the MESCAL project for negotiating QoS guarantees (e.g. bandwidth and delay) among ASes [10].

The model is based on two concepts: (1) negotiation of bandwidth offers between ASes; (2) establishment of unidirectional SLAs between ASes for the agreed bandwidth.

The key idea of the cascaded model is as follows. An AS offers bandwidth guarantees to its upstream ASes; each bandwidth offer specifies the reachable remote destination(s), the available bandwidth (e.g. maximum offered bandwidth) and a cost, for example, per unit of bandwidth. These destinations are either in customer ASes or reachable through downstream ASes. An upstream AS in general receives multiple bandwidth offers for any given destination, and has to decide which one to accept. Each accepted bandwidth offer is then established as a unidirectional SLA. The AS can then in turn make bandwidth offers to its upstream ASes, by combining its local bandwidth capabilities with the SLA. This process continues in a cascaded manner for further upstream ASes, and an end-to-end SLA chain can be built, with each SLA relying on the SLAs between downstream ASes.

Fig. 1 illustrates an example. Let *o-BW1* be the bandwidth guarantee offered by AS1 towards destination 'dest'. AS2 receives this offer *o-BW1*. We assume that AS2 decides to accept the bandwidth offer: AS2 then establishes an SLA with AS1 (*SLA2-1*) for this bandwidth. Now AS2 has a bandwidth guarantee provided by AS1 for access to 'dest'. AS2 can in turn extend this bandwidth guarantee by concatenating its local bandwidth capability with *SLA2-1*, and then offering a bandwidth (*o-BW2*) to AS3. *o-BW2* is the minimum of (a) the local bandwidth capability that AS2 is prepared to guarantee across its network and (b) *SLA2-1*. Now *o-BW2* indicates the bandwidth guarantee from AS2 to destination 'dest'. AS3 receives *o-BW2* from AS2 and it in turn repeats the decision process, possibly purchasing the offered bandwidth and establishing *SLA3-2*. In summary, once offers from other adjacent downstream ASes have been agreed as SLAs, an INP may build new extended services upon cascaded existing ones.

The decision on which bandwidth offers to accept, and how to effectively utilize the established SLAs and the AS′ intra-AS resources is non-trivial. In the next section, we propose a network dimensioning system, incorporating TE mechanisms, to solve this problem and make the best decisions.

4. Decomposition of the network dimensioning system

In this paper, we consider two problems, an economic and an engineering one, that need to be solved for provisioning end-to-end bandwidth guarantees. First, an AS needs to determine the appropriate amount of bandwidth to be purchased from each adjacent downstream AS so that the total bandwidth cost is minimized. Second, given these available bandwidth resources defined in the SLAs and the local network's bandwidth, the AS has to determine how to assign routes to the supported traffic in order to satisfy their bandwidth requirements while at the same time optimizing network resource utilization. We illustrate in Fig. 2 a decomposition of a network dimensioning system which consists of several components. We envisage this system as being offline and running infrequently as part of a resource provisioning cycle, for example, in the order of weeks.

4.1. Components of the network dimensioning system

The proposed network dimensioning system consists of the following components:

Inter-AS traffic forecast predicts inter-AS traffic in the network for a period of time [21,22] and records this information in an inter-AS Traffic Matrix (TM). Each element in the inter-AS TM is the aggregate traffic load that enters the network at an ingress point[1] and is destined for a remote destination prefix. The TM entry is represented by the tuple

`<ingress point, remote destination prefix, long-term average traffic demand>`

Some known methods can be used to compute the traffic aggregate, such as the effective bandwidth approach [23] if the mean and peak rates of the traffic are known.

The inter-AS TM is an important element for network and traffic engineering. Whilst an accurate inter-AS TM could be obtained through fine-grained flow-level traffic measurement this is not suitable for long-term predictions [2]. Nevertheless, these problems have recently been addressed with a methodology that allows an inter-AS TM to be predicted through measurement [24] and estimation for web traffic [25]. Alternatively, an inter-AS TM can be extrapolated from customer SLAs.

Inter-AS bandwidth discovery discovers bandwidth offers from adjacent downstream ASes through offline techniques, for example, advertisement. A bandwidth offer is uniquely identified by a connection point at which the offer is provided. Bandwidth offers are provided by adjacent ASes, and so the connection point, or inter-AS link on which it is offered, uniquely identifies the adjacent AS.

Each bandwidth offer specifies a maximum bandwidth towards a remote destination prefix and is associated with a cost, for example, per unit of bandwidth. Each bandwidth offer is represented by the tuple

`<egress router, adjacent AS border router address, remote destination prefix, maximum offered bandwidth, cost>`

Inter-AS bandwidth provisioning (IBP) addresses the economic problem described in the beginning of this section. For the sake of service resilience and load balancing [17], an increasing number of ASes have multiple connections to adjacent downstream ASes. As a result, an AS may receive multiple offers to each destination prefix from dif-

[1] Given the facts that INPs can generally only suggest which ingress peering points to use and that final decisions are still made by their customers [16], we assume in this paper that the ingress point of traffic is known in advance.

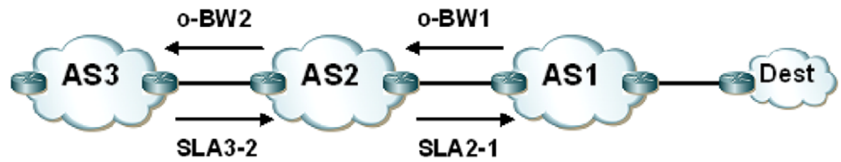


Fig. 1. Cascaded negotiation model.

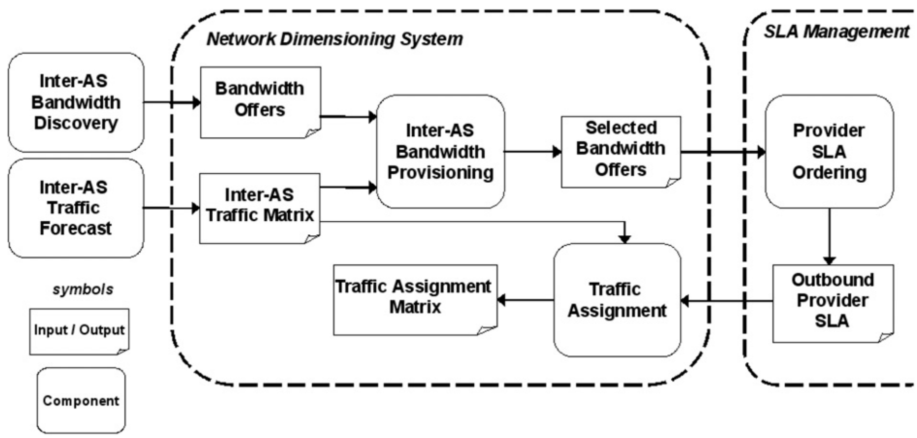


Fig. 2. Decomposition of the network dimensioning system.

ferent adjacent downstream ASes. The goal of IBP is to take as input the inter-AS TM and a set of bandwidth offers, and to produce as output a decision on which bandwidth offers to accept and the amount of bandwidth to be purchased from each of the accepted offers. Based on the IBP outcome, the AS will then establish SLAs (in this paper called outbound provider SLAs) with the adjacent downstream ASes to contract the bandwidth guarantees. We assume that the establishment of outbound provider SLAs is performed by the component "provider SLA ordering", a process whose details are outside the scope of this paper.

Traffic assignment (TA) deals with the engineering problem described in the beginning of this section. The goal of TA is to take as input an inter-AS TM, a set of outbound provider SLAs that are established after the IBP phase, and the available bandwidth resources of the AS, i.e. intra- and inter-AS link capacities, and then to assign appropriate routes for the supported traffic so that the bandwidth requirements are met while at the same time network resource utilization is optimized. An assignment of the route includes selection of an outbound provider SLA, an inter-AS link and an intra-AS route for the supported traffic. The key output of the TA is a Traffic assignment matrix that records the outbound provider SLAs, inter-AS links and intra-AS routes that have been selected for the supported traffic. Based on this matrix, an INP can implement the TA solution by configuring the network accordingly.

4.2. Inter-AS bandwidth overprovisioning

We can employ overprovisioning in the IBP phase. This implies that some network resources are left unused so as to protect the core backbone from failures and to accommodate some degree of traffic demand fluctuation [26]. Overprovisioning is also the current solution adopted by some INPs for QoS provisioning within their networks. For these reasons, we consider a certain amount of inter-AS bandwidth overprovisioning in this paper. During the IBP phase, the AS should not merely purchase bandwidth that marginally accommodates the forecasted traffic demand, because the bandwidth guarantee may not be maintained if even a small traffic upsurge occurs. A solution to this is to purchase more bandwidth than the forecasted traffic demand in order to insure against such traffic fluctuations. This also provides a buffer against inter-AS link failures, which may cause traffic to be shifted from one outbound provider SLA to another.

The task of IBP is thus to decide an appropriate amount of bandwidth to be purchased from the adjacent downstream ASes by taking into account overprovisioning. To do so, we introduce an overprovisioning factor $f_{\text{over}} \geqslant 1.0$ to specify the degree of inter-AS bandwidth overprovisioning. In principle, this factor is determined by considering the network's traffic characteristics and the target link utilization. However, since optimization of f_{over} is not the subject of this paper, we assume that a single value is used to represent the optimal overprovisioning that has already been determined by the ASes. The concept of overprovisioning factor has also been used by other researchers, for example [26].

Inter-AS bandwidth overprovisioning is implemented as follows. If $t(i,k)$ denotes the average demand of an inter-AS traffic flow aggregate, we define an inflated traffic flow, $t'(i,k) = t(i,k) \cdot f_{\text{over}}$.

5. Optimal inter-AS bandwidth provisioning

In this section and the next, we present the problem statement, formulation and algorithms of both the IBP and the TA problems.

Fig. 3 illustrates an AS topology with the key elements of the IBP problem. A set of border routers is connected to adjacent ASes. An ingress (or egress) router is the border router that receives (or sends) traffic from (or to) an adjacent AS. Each border router is associated with one or more inter-AS links. Each bandwidth offer is associated with a single inter-AS link. Each border router may receive multiple bandwidth offers for a remote destination prefix from different adjacent downstream ASes through different attached inter-AS links, for example, the top left border router in Fig. 3. Each inter-AS traffic flow enters the AS through a designated ingress router.

We define the total inter-AS bandwidth provisioning cost to be the total charge an AS pays for purchasing bandwidth from its adjacent downstream ASes. In this paper, we consider both linear and concave charge cost functions to reflect the effects of economies of scale in the pricing of inter-AS bandwidth capacity. The inter-AS bandwidth provisioning problem can be summarized as follows:

Given a set of bandwidth offers from adjacent downstream ASes, an inter-AS traffic matrix and a physical network topology, determine an appropriate amount of bandwidth to be purchased from each bandwidth offer so that the total inter-AS bandwidth provisioning cost is minimized while respecting the capacity constraints of the inter-AS links.

In solving the IBP problem we assume that the inter-AS traffic is non-splittable. This method not only can determine the appropriate amount of bandwidth to be purchased but also ensures that each traffic flow will be accommodated by at least one SLA during TA without causing the traffic to be split.

Note that some types of ASes, such as tier 2 and 3, may have both peering and customer–provider connections with adjacent ASes. A peering connection between two ASes refers to the case where each AS carries a similar amount of customer traffic from the other AS for free. On the other hand, a customer–provider connection refers to the case where the provider charges the customer for carrying traffic across its network. The IBP description in Section 4 assumed that an AS has only customer–provider connections with its adjacent downstream ASes and that a cost is associated with each bandwidth offer. In fact, peering connections can also be considered by IBP. In this case, the cost of bandwidth is typically zero and the maximum bandwidth will represent the agreed amount of traffic to be exchanged.

5.1. Inter-AS bandwidth provisioning problem formulation

We formulate IBP as an integer-programming problem. Table 1 shows the notation used throughout this paper. The objective of the IBP problem is to minimize the total IBP cost:

$$\text{Minimize} \sum_{i\in I}\sum_{k\in K}\sum_{oBw(k,j,n)\in Out(k)} x_{i,k}^{j,n}\cdot chg_k^{j,n}\cdot t'(i,k) \tag{1}$$

subject to:

$$\sum_{i\in I}\sum_{k\in K} x_{i,k}^{j,n}\cdot t'(i,k)\leqslant c_{\text{inter}}^{j,n}\quad \forall (j,n)\ \text{where}\ j\in J,\ n\in NEXT_j \tag{2}$$

$$\sum_{i\in I} x_{i,k}^{j,n}\cdot t'(i,k)\leqslant MaxBw_k^{j,n}\quad \forall (k,j,n)\ \text{where}\ k\in K,\ j\in J,\ n\in NEXT_j \tag{3}$$

$$x_{i,k}^{j,n}, y_k^{j,n}\in\{0,1\} \tag{4}$$

$$x_{i,k}^{j,n}\leqslant y_k^{j,n}\quad \forall (i,k,j,n)\ \text{where}\ i\in I,\ k\in K,\ j\in J,\ n\in NEXT_j \tag{5}$$

$$\sum_{oBw(k,j,n)\in Out(k)} x_{i,k}^{j,n}=1\quad \forall (i,k)\ \text{where}\ i\in I,\ k\in K \tag{6}$$

$$\sum_{n\in NEXT_j} y_k^{j,n}\leqslant 1\quad \forall (k,j)\ \text{where}\ k\in K,\ j\in J \tag{7}$$

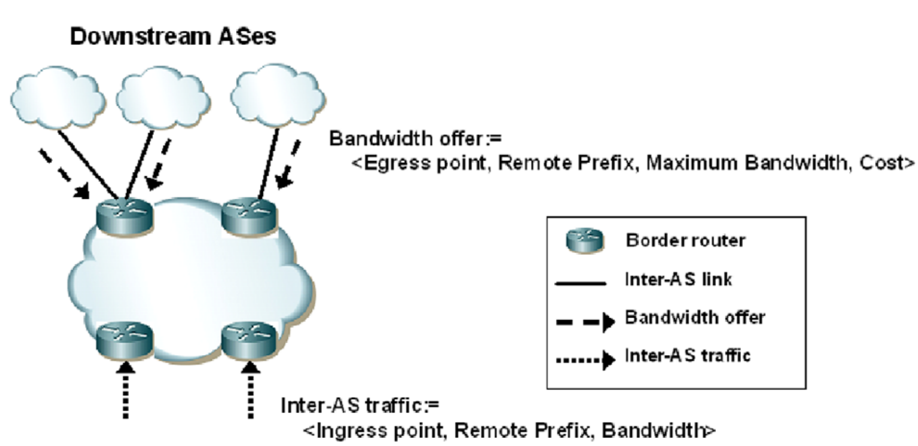


Fig. 3. Elements of the inter-AS bandwidth provisioning problem.

Table 1
Notation used in this paper

Notation	Description
General notation	
K	A set of destination prefixes
I	A set of ingress routers
J	A set of egress routers
f_{over}	Overprovisioning factor
$t(i,k)$	Bandwidth demand of an inter-AS traffic flow entering the AS at ingress router $i \in I$ heading towards destination prefix $k \in K$. It is considered by the TA problem
$t'(i,k)$	Inflated traffic flow $t(i,k)$. It is considered by the IBP problem
$Out(k)$	A set of bandwidth offers that has reachability to destination prefix k
$NEXT_j$	A set of next-hop addresses (addresses of the border routers in adjacent downstream ASes) that is associated with egress router $j \in J$
$C_{\text{inter}}^{j,n}$	Capacity of the inter-AS link that connects egress router j to next-hop address $n \in NEXT_j$
$bw_{\text{inter}}^{j,n}$	Residual bandwidth of $C_{\text{inter}}^{j,n}$
C_{intra}^{l}	Capacity of intra-AS link l
bw_{intra}^{l}	Residual bandwidth of C_{intra}^{l}
Notation used in IBP	
$oBw_k^{j,n}$	A bandwidth offer that is associated with destination prefix k and is advertised through the inter-AS link that connects egress router j to next-hop address n
$MaxBw_k^{j,n}$	Maximum bandwidth of the offer $oBw_k^{j,n}$
$Chg_k^{j,n}$	A charge per unit bandwidth for $oBw_k^{j,n}$
$x_{i,k}^{j,n}$	Variable indicating whether traffic flow $t'(i,k)$ is assigned to bandwidth offer $oBw_k^{j,n}$
$y_k^{j,n}$	Variable indicating whether the bandwidth offer $oBw_k^{j,n}$ is selected
Notation used in TA	
$pSLA_k^{j,n}$	Outbound provider SLA of the bandwidth offer $oBw_k^{j,n}$
$pSLAC_k^{j,n}$	Contracted bandwidth specified in outbound provider SLA $pSLA_k^{j,n}$
$pSLABw_k^{j,n}$	Residual bandwidth of $pSLAC_k^{j,n}$
$dist_{i,j}^{k}$	Number of hops on the intra-AS route between ingress router i and egress router j towards destination prefix k
$P_{i,j}$	A set of feasible intra-AS routes between ingress router i and the egress router j to which the selected outbound provider SLA is associated
$w_{i,k}^{p}$	Variable indicating whether path $p \in P_{i,j}$ is chosen to realize the traffic flow $t(i,k)$
$z_{i,k}^{j,n}$	Variable indicating whether traffic flow $t(i,k)$ is assigned to outbound provider SLA $pSLA_k^{j,n}$
$\Upsilon_{i,k}^{l}$	Variable indicating whether traffic flow $t(i,k)$ is assigned to intra-AS link l

Constraint (2) ensures that no inter-AS link carries traffic exceeding its capacity. Constraint (3) ensures that no bandwidth offer carries traffic exceeding its maximum capacity. Constraint (4) ensures that the discrete variables assume binary values. Constraint (5) ensures that, whenever traffic flow $t'(i,k)$ is assigned to bandwidth offer $oBw_k^{j,n}$, then this bandwidth offer must have been selected. Constraint (6) ensures that only one bandwidth offer is selected for each inter-AS traffic flow. Hence, traffic splitting over multiple bandwidth offers is not considered in this paper. Constraint (7) ensures that only one of the bandwidth offers, which are advertised at a border router through different inter-AS links, is selected for each remote destination prefix. This constraint ensures the BGP rule that only one route toward a remote destination prefix is selected as the best route. This makes the IBP implementation easier through BGP configuration.

5.2. Modified inter-AS bandwidth provisioning problem

We assume that when multiple bandwidth offers towards the same remote destination prefix k are present at a given border router j (i.e. $\exists n\ MaxBw_k^{j,n} > 0$), the AS has already determined the best one as a candidate bandwidth offer. Thus, each border router will consider at most one bandwidth offer towards each remote destination. The decision of selecting the best bandwidth offer might be based on business factors such as the relationships between ASes and the reputations of adjacent downstream ASes. As a result of this assumption, the variable $y_k^{j,n}$ of which bandwidth offer has been considered for each remote destination prefix k at each border router j is pre-determined and this satisfies constraint (7) since at most one bandwidth offer will be considered (i.e. $\sum_{n \in NEXT_j} y_k^{j,n} \leqslant 1$). Therefore, constraint (7) is automatically enforced. Nevertheless, the revised IBP problem, which consists of Eqs. (1)–(6) is still challenging because of its NP-hardness, which we show in the next section.

5.3. NP-hardness of the inter-AS bandwidth provisioning problem

In this section, we show that the inter-AS bandwidth provisioning problem is NP-hard [45]. The proof given in [16] has shown that the inter-AS TE optimization problem, which consists of analogous objective function (1), constraints (2) and (6), is NP-hard by mapping the problem to the Generalized Assignment Problem (GAP) which is known to be NP-hard. Given the additional constraints (3)–(5) and (7) on the inter-AS TE optimization problem, the IBP problem is therefore also NP-hard. More specifically, as two types of resource (i.e. capacities of inter-AS link and outbound provider SLA) are consumed for traffic flows, the IBP problem can also be shown to be an extended version of the GAP, Multi-Resource Generalized Assignment Problem, which is known to be NP-hard [27].

5.4. A lower bound of the inter-AS bandwidth provisioning problem

Since the IBP problem is NP-hard, it is not in general possible to efficiently find an optimal solution in order to compare with the performance of our proposed IBP algorithms. Therefore, we need to derive an approximated optimal solution that can be obtained efficiently by relaxing some constraints. This approximated optimal solution is thus a lower bound of the IBP problem. A lower bound typically has better result than the optimal solution because some problem constraints are relaxed. However, due to the relaxation, it is not a valid solution to the problem. Nevertheless, the lower bound is a good approximation of an optimal solution for heuristic algorithms to compare their performance. We show the derivation of a lower bound for the IBP problem as follows.

We derive a lower bound by relaxing some IBP problem constraints. First of all, constraint (7) is automatically enforced by our assumption that each border router has only considered the best candidate bandwidth offer towards each remote destination prefix. Second, we relax the non-bifurcation integer constraint (4). In many practical situations, integer-programming problems, which require all variables to be integers, are NP-hard. Instead, a linear programming problem that has only non-integer variables can be generally solved efficiently in the worst case. Therefore, we relax constraint (4) to

$$0 \leqslant x_{i,k}^{j,n} \leqslant 1, \text{ non-integer} \tag{8}$$

Finally, we find that a lower bound can be readily calculated by the following method if inter-AS link capacity constraint (2) is relaxed. Relaxation of a capacity constraint means that the constraint is simply ignored based on the assumption that capacity is large enough to accommodate the traffic.

Given $Pr_{\text{low}} = \underset{\forall k,j,n}{Min}\ Chg_k^{j,n}$ and

$Pr_{\text{high}} = \underset{\forall k,j,n}{Max}\ Chg_k^{j,n}$, we can define

$$b_k^{\psi} = \sum_{j\in J} \sum_{n\in Next_j} MaxBw_k^{j,n} | Chg_k^{j,n} = \psi \quad \forall Pr_{\text{low}} \leqslant \psi \leqslant Pr_{\text{high}} \quad \text{and} \quad k \in K \tag{9}$$

where $b_k^{\psi} \geqslant 0$ is the sum of maximum capacity of all the bandwidth offers to remote destination prefix k with a charge equal to ψ, and

$$d_k = \sum_{i\in I} t'(i,k) \quad \forall k \in K \tag{10}$$

where $d_k \geqslant 0$ is the sum of bandwidth demands of all the traffic flows to remote destination prefix k.

For each traffic demand d_k towards remote destination prefix k, we first attempt to assign it to the lowest cost bandwidth offer. If the lowest cost bandwidth offer cannot entirely accommodate the traffic demand due to capacity limitation, then the residual demand will be assigned to the next lowest cost bandwidth offer. This traffic demand assignment iterates until the bandwidth offer with a particular cost can entirely accommodate the traffic demand. A lower bound is calculated based on the traffic assigned to each bandwidth offer and its associated cost. A lower bound, using the above-mentioned method, can be calculated by

$$\sum_{k\in K} \sum_{\Psi=Pr_{\text{low}}}^{Pr_{\text{high}}} \left\{ Min\left[Max\left(d_k - \sum_{\alpha=Pr_{\text{low}}}^{\Psi-1} b_k^{\alpha}, 0 \right), b_k^{\Psi} \right] \cdot \Psi \right\} \tag{11}$$

For a particular cost ψ, the *max* function determines the residual traffic demand that has not been allocated to the bandwidth offers that have lower cost than the one being considered. The *min* function attempts to assign this residual traffic demand to the bandwidth offer with the cost currently being considered. The inner summation symbol considers all bandwidth offers toward a remote destination prefix with different costs. The outer summation symbol considers all the remote destination prefixes. Hence, a lower bound can be computed efficiently.

5.5. A genetic algorithm

In the previous section, we proved the IBP problem to be NP-hard. We therefore propose an efficient genetic algorithm (GA) to obtain a near-optimal solution of the IBP problem. Genetic algorithm (see Fig. 4) [28] is an algorithm that operates by the natural selection of 'survival of the fittest'. It has been successful in solving many large-scale optimization problems. More details on GA can be found in [28,47].

To solve the IBP problem, we modify and extend the GA [29] proposed for solving the Generalized Assignment Problem [30]. The steps of our GA are described as follows:

Step 1. Create a feasibility mapping table which maps all the feasible bandwidth offers to each inter-AS traffic flow. A bandwidth offer $oBw_k^{j,n}$ is feasible for an inter-AS traffic flow $t'(i,k)$ if the following constraints are satisfied:

- $oBw_k^{j,n} \in Out(k)$ (12)
- $t'(i,k) \leqslant c_{\text{inter}}^{j,n}$ (13)
- $t'(i,k) \leqslant MaxBw_k^{j,n}$ (14)

Constraint (12) ensures that the remote destination prefix in the bandwidth offer matches the requested remote destination prefix of the traffic flow. Constraints (13) and (14) ensure, respectively, that the bandwidth demand of the traffic flow does not exceed the capacity of either the inter-AS link to which the bandwidth offer is associated or the maximum capacity of the bandwidth offer. These constraints, however, do not guarantee that constraints (2) or (3) are met for the entire chromosome.

Step 2. Generate an initial population of C randomly constructed chromosomes. Fig. 5 shows a representation of an individual chromosome which consists of T genes where T is the number of inter-AS traffic flows and each gene represents an assignment between a traffic flow and a bandwidth offer. The identifier given to each traffic flow represents each inter-AS traffic flow $t'(i,k)$. Let $S_{t'(i,k),c} = \langle k,j,n \rangle$ represent the bandwidth offer $oBw_k^{j,n}$ that has been assigned to traffic flow $t'(i,k)$ in chromosome $c \in C$. Each gene of the initial chromosomes is generated by randomly assigning a feasible bandwidth offer to each traffic flow according to the feasibility mapping table produced in step 1. Note that an initial chromosome may not be a feasible solution since the capacity constraint (2) or (3) could be violated.

Step 3. Decode each chromosome to obtain its fitness value. The fitness of chromosome c is equal to the total inter-AS bandwidth provisioning cost, given by

$$-\sum_{i\in I}\sum_{k\in K} Chg(s_{t'(i,k),c}) \cdot t'(i,k) \tag{15}$$

The negative sign reflects the fact that a solution with lower cost has higher fitness. We define $Chg(s_{t'(i,k),c}) \cdot t'(i,k)$ to be the IBP cost for the traffic flow $t'(i,k)$. If the chromosome contains an infeasible solution, a common approach is to penalize its fitness for the infeasibility. Instead of this, we adopt the approach in [29] and associate an unfitness value for each chromosome. The unfitness value of chromosome c is the degree of infeasibility of the chromosome, which equals the amount of violated capacity summed over all the inter-AS links and all the bandwidth offers,

$$\sum_{j\in J}\sum_{n\in Next_j} Max\left\{0, \sum_{i\in I,k\in K: S_{t'(i,k),c}=\langle k,j,n\rangle} t'(i,k) - c_{\text{inter}}^{j,n}\right\} + \sum_{k\in K}\sum_{j\in J}\sum_{n\in Next_j} Max\left\{0, \sum_{i\in I,: S_{t'(i,k),c}=\langle k,j,n\rangle} t'(i,k) - MaxBw_k^{j,n}\right\} \tag{16}$$

```
Procedure GA
{
    initialize population;
    while termination condition not satisfied
    do
    {
        evaluate the fitness of each chromosome
        select chromosomes for reproduction
        apply genetic operators (crossover and mutation) to create child chromosomes
        replace unfit chromosomes with child chromosomes
    }
}
```

Fig. 4. Genetic algorithm.

Traffic flow	1	2	···	T-1	T
Bandwidth offer	o-BW1	o-BW2	···	o-BWm	o-BWn

Fig. 5. Representation of an individual's chromosome.

With the separation of fitness and unfitness values, chromosomes can be evaluated in a two-dimensional plane, so the selection and replacement can direct the search towards feasible solutions by replacing highly unfit chromosomes with lightly unfit or entirely fit chromosomes.

Step 4. Select two parent chromosomes for reproduction. We use the pairwise tournament selection method. In pairwise tournament selection, two individual chromosomes are chosen randomly from the population and the one that is fitter (higher fitness value) is selected for a reproductive trial. Two pairwise tournament selections are held, each of which produces one parent chromosome, in order to produce a child chromosome.

Step 5. Generate two child chromosomes by applying a simple one-point crossover operator on the two selected parents. The crossover point p_{co} is randomly selected. The first child chromosome consists of the first p_{co} genes from the first parent and the remaining $(n - p_{co})$ genes from the second parent. The second child chromosome takes the parent genes that have not been considered by the first child chromosome.

Step 6. Perform a probabilistic mutation on each child chromosome. The mutation simply exchanges elements in two selected genes (i.e. exchange the assigned bandwidth offers between two randomly selected traffic flows) without violating constraints (12)–(14).

Step 7. The fitness and unfitness values of child chromosomes can be improved by applying the following two problem-specific heuristic operators:

- *Heuristic-A*: For each inter-AS traffic flow that has been assigned to an infeasible bandwidth offer such that either capacity constraint (2) or (3) is violated, find a feasible bandwidth offer that incurs the lowest IBP cost for the traffic flow. Denote $\Delta t'(i,k)$ the difference between the original IBP cost induced by the traffic flow and the new IBP cost after the traffic flow has been reassigned to a feasible bandwidth offer. Among those inter-AS traffic flows, select the one with the lowest $\Delta t'(i,k)$ and assign it to the corresponding selected feasible bandwidth offer. This heuristic operator iterates at most H times where H is a parameter that optimizes the algorithm's performance or stops when no inter-AS traffic flows have been assigned to infeasible bandwidth offers.

- *Heuristic-B*: For each inter-AS traffic flow, find a feasible bandwidth offer that produces the lowest IBP cost. If such a feasible bandwidth offer has been found, reassign the traffic flow to it.

Heuristic-A aims to reduce the unfitness value of the child chromosome by reassigning traffic flows from infeasible to feasible bandwidth offers while keeping the total IBP cost as low as possible. *Heuristic-B* attempts to improve the fitness of the child chromosome by reassigning traffic flows to feasible bandwidth offers with lower costs.

Step 8. Replace two chromosomes in the population by the improved child chromosomes. In our replacement scheme, chromosomes with the highest unfitness are always replaced by the fitter child chromosomes. If no unfit solution exists, the lowest fitness ones are replaced.

Step 9. Repeat steps 4–8 until N_{cd} child chromosomes have been produced and placed in the population.

Step 10. Check if the GA termination criterion is met. The termination criterion is that either both the average and the best fitness over all the chromosomes in the two consecutive generations are identical or once the selected number of iterations, N_{it}, has been reached in order to avoid excess algorithm execution time. Steps 4–9 iterate until the termination criterion is met.

5.6. Time complexity of the proposed genetic algorithm

In this section, we analyze the time complexity of the proposed GA.

Theorem 1. *The worst-case time complexity of the GA is* $O(N_{it} \cdot N_{cd} \cdot Hnm)$.

Proof. Denote by m the number of bandwidth offers, n the number of traffic flows and c the number of chromosomes. The creation of the feasibility mapping table in step 1 of the GA could be done in $O(mn)$ time. The generation of chromosomes in step 2 could be done in $O(cn)$ time. The decoding of chromosomes in step 3 could be done in $O(cn)$ time. The small worst-case time complexity of light-weight steps 4–6 can be simply ignored. The problem-specific *Heuristic-A* requires $O(Hnm)$ time while *Heuristic-B* requires $O(mn)$ time. Since each child chromosome is decoded followed by the two heuristic operations, the entire step 7 requires $O(n + Hnm + mn)$ time. Since N_{cd} child chromosomes are produced, the iteration in step 9 requires $O(N_{cd} \cdot (n + Hnm + mn))$ time. The GA runs until the termination criterion in step 10 is met, which takes at most N_{it} iterations. Compared to the high time complexity of the iteration from step 4 to step 10, the time complexity from step 1 to step 3 could be ignored. After summarizing and simplifying the above analysis, the overall worst-case time complexity of the GA is $O(N_{it} \cdot N_{cd} \cdot Hnm)$. □

6. Optimal traffic assignment

Let us assume that the bandwidth offers selected by the IBP (Section 5) have now been accepted and configured as a set of outbound provider SLAs. Given this set and the available bandwidth capacity within the AS, we now consider how to assign routes to the traffic so as to meet the traffic's bandwidth requirements. Fig. 6 shows that from the viewpoint of AS-1, a route to the destination can be decomposed into three parts: (1) the intra-AS route, (2) the inter-AS link and (3) the inter-AS route from the downstream AS (AS-2) to the destination AS (AS-3). Sufficient bandwidth must be provisioned in all parts of this route in order to satisfy the bandwidth demand. Once the outbound provider SLA is known, the available bandwidth resource on any part of the route is known to the AS: the intra- and inter-AS links are owned by the AS and the available bandwidth from the downstream AS to the destination AS is guaranteed by the outbound provider SLA. As a result, the TA problem can be defined as follows:

> *Given a set of outbound provider SLAs, an inter-AS TM and a physical network topology, assign end-to-end routes to the supported traffic so that the bandwidth requirement is satisfied while optimizing network resource utilization. A route assignment includes the selection of an outbound provider SLA, an inter-AS link and an explicit intra-AS route from the ingress router to the egress router where the selected outbound provider SLA is associated.*

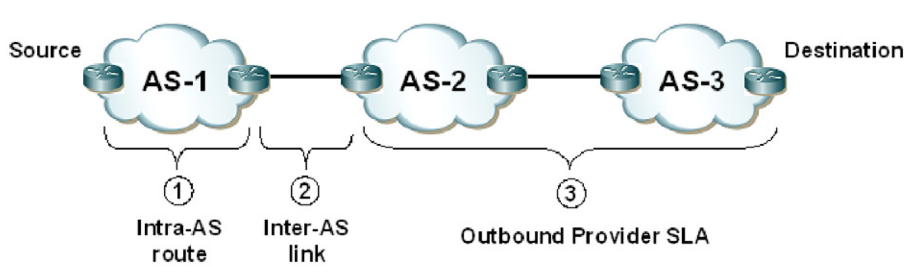


Fig. 6. Essential components for end-to-end bandwidth guarantee.

In this paper, we assume that explicit intra-AS routes are implemented by MPLS. In addition, there are many optimization criteria for network resource utilization, such as minimizing resource consumption or load balancing. For simplicity, the network resource utilization used in this paper is a general metric, the total bandwidth consumed in carrying traffic across the network. The reader is also referred to [31] for a multi-objective TA problem.

6.1. Traffic assignment problem formulation

As with the IBP problem of Section 5, we formulate the TA problem as an integer-programming problem. The fundamental objective is to provide bandwidth guarantees to inter-AS traffic by satisfying their bandwidth demands. We define the bandwidth demand of an inter-AS traffic flow $t(i,k)$ to be met if the following constraints are satisfied:

- There exists at least one feasible path $f_{\text{path}} \in P_{i,j}$ from ingress router i to egress router j to which the selected outbound provider SLA is associated, i.e.
$$Min_{\forall l \in f_{\text{path}}} bw^{l}_{\text{intra}} \geqslant t(i,k) \quad (17)$$
- $$bw^{j,n}_{\text{inter}} \geqslant t(i,k) \quad (18)$$
- $$pSLABw(k,j,n) \geqslant t(i,k) \quad (19)$$

Constraint (17) ensures that there exists at least one feasible path between the ingress point and the selected egress point, and the bottleneck bandwidth of the path is not less than the bandwidth demand of the traffic flow. Constraints (18) and (19) ensure that the inter-AS link and the outbound provider SLA, respectively, have sufficient bandwidth to accommodate the traffic flow.

The objective of minimizing the total bandwidth consumption within the network can be translated to the problem of minimizing the total number of hops that a traffic flow must traverse in the network, i.e.

$$\text{Minimize} \sum_{i\in I}\sum_{k\in K}\sum_{oBw(k,j,n)\in Out(k)} z^{j,n}_{i,k}\cdot dist^{k}_{i,j}\cdot t(i,k) \quad (20)$$

subject to:

$$\sum_{i\in I}\sum_{k\in K} z^{j,n}_{i,k}\cdot t(i,k) \leqslant c^{j,n}_{\text{inter}} \quad \forall (j,n) \text{ where } j\in J,\ n\in NEXT_j \quad (21)$$

$$\sum_{i\in I}\sum_{k\in K} \Upsilon^{l}_{i,k}\cdot t(i,k) \leqslant c^{l}_{\text{intra}} \quad \forall l\in E \quad (22)$$

$$\sum_{i\in I} z^{j,n}_{i,k}\cdot t(i,k) \leqslant pSLAC^{j,n}_{k} \quad \forall (k,j,n) \text{ where } k\in K,\ j\in J,\ n\in NEXT_j \quad (23)$$

$$z^{j,n}_{i,k}, \Upsilon^{l}_{i,k}, w^{p}_{i,k} \in \{0,1\} \quad (24)$$

$$\sum_{oBw(k,j,n)\in Out(k)} z^{j,n}_{i,k} = 1 \quad \forall (i,k) \text{ where } i\in I,\ k\in K \quad (25)$$

$$\sum_{p\in P_{i,j}} w^{p}_{i,k} = 1 \quad \forall (i,k) \text{ where } i\in I,\ k\in K \quad (26)$$

$$\Upsilon^{l}_{i,k} \leqslant w^{p}_{i,k} \quad \forall (l,p,i,k) \text{ where } l\in p,\ p\in P_{i,j},\ i\in I,\ k\in K \quad (27)$$

Constraints (21)–(23) ensure that the total traffic assigned to the inter-AS link, the intra-AS link and the outbound provider SLA do not exceed their respective capacities. Constraint (24) ensures the discrete variables assume binary values. Constraint (25) ensures that only one outbound provider SLA is selected for each traffic flow. Constraint (26) ensures that each traffic flow $t(i,k)$ is routed along a single intra-AS route in order to preserve scalability and minimize network management complexity. Constraint (27) ensures that, whenever traffic flow $t(i,k)$ is assigned to intra-AS link l, then the path to which l is associated must have been selected. Moreover, given the lossless property of the links, an additional constraint that has not been presented is the flow conservation constraint which ensures that the traffic flowing into a node must equal the traffic flowing out of the node for any intermediate node.

Although the TA problem formulation has been presented, we find that it can be simplified by neglecting the inter-AS link capacity constraint.

Lemma 1. *Given that the IBP phase has been successfully completed, the inter-AS link capacity constraint can be neglected, since the outbound provider SLA bandwidths satisfy the inter-AS link capacity constraint.*

Proof. An inter-AS link (j,n) is considered. According to constraints (2) and (3) of the IBP problem, the total bandwidth purchased from the bandwidth offers (i.e. the outbound provider SLAs) that are associated with the inter-AS link must not exceed the link capacity, i.e

$$\sum_{k\in K} pSLABw^{j,n}_{k} \leqslant c^{j,n}_{\text{inter}} \quad (28)$$

Eq. (28) concludes that the inter-AS link capacity constraint is satisfied if the outbound provider SLA constraint is satisfied. This conclusion applies to all the inter-AS links and the outbound provider SLAs associated with the links. Hence, the inter-AS link capacity constraint (21) of the TA problem can be simply neglected as long as the outbound provider SLA bandwidth constraint (23) is considered. Given the condition that each inter-AS link is associated with only one outbound provider SLA for each destination prefix, the selection of outbound provider SLAs reflect that the associated inter-AS links will also be automatically selected. □

6.2. NP-hardness of the traffic assignment problem

In this section, we show that the TA problem is NP-hard. As introduced at the beginning of Section 6, the TA problem consists of two sub-problems: (1) optimal selection of outbound provider SLA and the associated inter-AS link; (2) optimal selection of intra-AS route. We assume that the first sub-problem is pre-determined. In other words, each traffic flow is pre-assigned to an outbound provider SLA. The second part of the TA problem, i.e. the optimal selection of intra-AS route, is to find an optimal route for each traffic flow from the ingress router to the egress router where the selected outbound provider SLA is associated. It has been shown in [46] that this intra-AS optimal routing problem can be mapped to the Multicommodity Flow Problem (MFP) [32]. Since the TA problem does not allow arbitrary traffic splitting, i.e. constraint (26), it is a version of the Integral MFP which is known to be NP-hard [32]. Hence, the TA problem is NP-hard.

6.3. A greedy heuristic algorithm for the traffic assignment problem

In comparing the two problems in the network dimensioning system, the complexity of the TA Problem is higher than the IBP problem, in terms of number of decision variables and constraints. In addition, the TA is performed more frequently than the IBP: network capacity expansion is usually less frequent than traffic engineering. Based on these reasons, the algorithm for solving the TA problem should be more *efficient* than the IBP algorithm. In general, a GA can produce a better performance but with higher time complexity than simple greedy-based heuristics. Due to the higher complexity of the TA problem, we do not consider using GA to solve the TA problem as we did for the IBP problem. Instead, we present a simple and efficient greedy heuristic algorithm to solve the TA problem, namely a greedy-penalty heuristic.

Greedy-penalty heuristic: It is possible that the order in which traffic flows are assigned to outbound provider SLAs may produce different selection results. For example, if we take a traffic flow $t(i,k) = 2$, we might assign it greedily to some outbound provider SLA $pSLA_k^{j,n}$ with intra-AS distance $dist_{i,j}^k = 3$. In this case, the total bandwidth consumed equals 6. If on the other hand we allocate it later in the process, the outbound provider SLA may not have sufficient bandwidth because its bandwidth has been allocated to other traffic flows and the considered traffic flow might have to be assigned to another outbound provider SLA $pSLA_k^{j',n'}$, for example, with $dist_{i,j'}^k = 6$. As a result, the total bandwidth consumed equals 12. In this case, we have a penalty on the consumption of additional bandwidth (i.e. $12 - 6 = 6$) and we use *penalty* to refer to this value. A penalty-based algorithm aims to minimize the number of hops a flow must traverse by placing customer traffic flows in certain order according to *penalty*. We propose a greedy-penalty heuristic algorithm that takes into consideration the penalty value. Such an algorithm has also been proposed to solve the GAP [30].

Step 1. For each unassigned traffic flow, we measure the desirability of assigning it to each feasible outbound provider SLA that satisfies constraint (19). The desirability is the total bandwidth consumed by the traffic flow along the intra-AS route between the ingress and the egress router with which the outbound provider SLA is associated (i.e. the number of intra-AS hops times the bandwidth demand). Intra-AS route computation is done by Constrained Shortest Path First (CSPF) [33], which finds a route that is shortest in terms of hop while satisfying the bandwidth requirement. The smaller the desirability, the smaller amount of bandwidth to be consumed, and thus the better the selection.

Step 2. Compute *penalty* for each unassigned traffic flow, being the difference between the desirability of the traffic flow's best and second best selection (i.e. the two outbound provider SLAs which yield the smallest desirability). If there is only one feasible outbound provider SLA with sufficient spare capacity to accommodate the traffic flow, we need to set *penalty* to infinity and immediately assign the traffic flow to it. Otherwise, this feasible outbound provider SLA may subsequently become unavailable, resulting in an invalid solution.

Step 3. Among all unassigned traffic flows, the one yielding the largest *penalty* is placed with its best selection. In other words, this traffic flow is assigned to the feasible outbound provider SLA that achieves the smallest desirability. If multiple traffic flows which have the same largest penalty exist, the one with the largest bandwidth demand is placed. If there are several such traffic flows, one is chosen randomly.

Step 4. Once the outbound provider SLA is selected, the requested bandwidth is allocated on the corresponding selected intra-AS route and the outbound provider SLA to establish an end-to-end bandwidth guaranteed route. We iterate step 1 to step 4 until all the traffic flows have been considered.

6.4. Time complexity of the greedy-penalty heuristic

In this section, we analyze the time complexity of the greedy-penalty heuristic.

Theorem 2. *The worst-case time complexity of the greedy-penalty heuristic algorithm is* $O(n^2m(|V|\log|V| + |E|))$.

Proof. The path computation of the CSPF can be implemented by first eliminating all intra-AS links that do not have sufficient bandwidth for a traffic flow and then running a Dijkstra-based minimum hop count algorithm on the remaining graph. The time complexity of a Fibonnaci-heap implementation of the Dijkstra algorithm is $O(|V|\log|V| + |E|)$ where $|V|$ and $|E|$ are the number of nodes and links in the network, respectively [34]. The desirability of assigning a traffic flow to an outbound provider SLA is known once the corresponding intra-AS route is found. The greedy-penalty heuristic algorithm then determines *penalty* for all the traffic flows. This requires $O(nm(|V|\log|V| + |E|))$ where n and m are number of traffic flows and number of egress routers, respectively. The algorithm assigns a traffic flow to an outbound provider SLA each time based on *penalty* and some tie-break decisions. Finally, the algorithm iterates until all traffic flows have been considered. Therefore, we conclude that the overall worst-case time complexity of the greedy-penalty heuristic algorithm is $O(n^2m(|V|\log|V| + |E|))$. □

7. Performance evaluation

We evaluate the proposed GA and the greedy-penalty heuristic algorithms by simulation. The simulation software was written in Java. The computation was carried out on a laptop with an Intel Pentium Centrino 1.5 GHz Processor with 512 MB RAM. All the results presented in this paper are an average of 50 different simulation trials.

7.1. Network model

We use a network topology generated by BRITE [35] with 100 nodes and average node degree of 4. These numbers were chosen to represent a medium to large INP topology. All intra-AS links are unidirectional and each has capacity of 500 units. Note that, since no realistic data is publicly available, we assume that the values of link capacity, bandwidth offers, and traffic demand are unitless. Therefore, these values that we use in this paper may represent any specific value depending on the definition of the corresponding unit.

Among the 100 nodes, 30 nodes are randomly selected as border routers and the remaining nodes are core routers. In practice, each border router may connect with several inter-AS links to adjacent ASes. However, for simplicity, and without loss of generality, we abstract these inter-AS links into one. Thus, each border router is associated with one virtual inter-AS link which can logically represent one or multiple physical inter-AS links. Therefore, 30 virtual inter-AS links are considered and each has capacity of 500 units.

7.2. Bandwidth offer model

It is well known that whilst a typical default-free routing table may contain routes for more than 100,000 prefixes, only a small fraction of prefixes are responsible for a large fraction of the traffic [15]. Based on this finding, we consider 100 remote destination prefixes to be included in the bandwidth offers. In fact, each of them may not merely represent an individual prefix but also a group of distinct address prefixes that have the same end-to-end path properties, for example, geographical location, offering AS and maximum available bandwidth. Hence, the hundred prefixes we considered could actually reflect an even larger number of prefixes.

In a network, each border router can be an ingress or egress point. Without loss of generality, we consider the network scenario where if a border router receives a bandwidth offer towards destination prefix k from adjacent AS Y, then AS Y cannot inject traffic for k into it. This corresponds to multi-hop traffic [21] in which the traffic traverses the network instead of being directed to another egress link of the same border router. We adopt this model in order to evaluate the TA objective of total bandwidth consumption in the network. As a result, we cannot assign all the destination prefixes on each border router as bandwidth offers. Instead, at each border router we randomly select half of these hundred destination prefixes as bandwidth offers and the other half as inter-AS traffic. In other words, we set the average number of distinct bandwidth offers advertised at each border router to be half of the number of prefixes. Furthermore, each border router can generate the number of traffic flows towards half of these prefixes that have not been selected for bandwidth offers. We note that this destination prefix generation process is just a best effort attempt to model prefix distribution, as no synthetic model for the actual behavior of prefix distribution in real networks was found in the literature. The remote destination prefixes associated with the bandwidth offers are randomly selected. The maximum capacity of each bandwidth offer is uniformly generated between 100 and 200 units. The charge associated with each bandwidth offer varies according to the simulation scenarios.

7.3. Traffic model

Ingress points and remote destination prefixes of the inter-AS traffic matrix are randomly generated. Previous work has shown that inter-AS traffic is not uniformly distributed [36]. According to [37], the AS traffic volumes are top-heavy and can be approximated by a Weibull distribution with shape parameter 0.2–0.3. We therefore generate the inter-AS TM with traffic demand following this distribution with the shape parameter 0.3. As previously mentioned, we do not allow traffic-prefix looping, so that if the AS receives a bandwidth offer towards remote destination prefix k from an adjacent AS, then this adjacent AS cannot inject traffic into the AS for k. The number of inter-AS traffic flows to be considered ranges from 500 to a maximum 1500.

As mentioned in Section 4.1, each inter-AS traffic flow is an aggregate of individual traffic flows that have identical ingress points and remote destination prefixes. Hence, the number of inter-AS traffic flows we considered does not reflect the exact total number of individual traffic flows. Instead, the number could represent more individual traffic flows. We assume that moderate overprovisioning is considered by the IBP and unless specified, $f_{\text{over}} = 1.25$ (i.e. 25% inter-AS bandwidth overprovisioning). Table 2 shows the number of traffic flows, their corresponding traffic volume and overall inter-AS link utilization. Note that the total traffic volume presented in the table has already taken into account the overprovisioning factor.

Table 2
Inter-AS traffic

Number of traffic flows	Total traffic volume	Overall inter-AS egress link utilization (%)
500	4465	30
625	5578	37
750	6719	45
875	7813	52
1000	8915	60
1125	10,046	67
1250	11,142	74
1375	12,259	82
1500	13,402	90

7.4. Algorithm parameters

For the IBP's GA parameters, we adopt the suggested values from previous GA research to achieve satisfactory effectiveness and convergence rate of the algorithm [38]. The population size is 200, the value of H of the heuristic operator (a) is 200 since the IBP problem is highly constrained by two capacity constraints, N_{cd} is set to 50, the probability of mutation is 0.01 and N_{it} is set to 100.

7.5. Evaluation of the IBP algorithms

We compare the performance of our proposed GA described in Section 5.5 with the following alternatives:

Greedy-cost heuristic: The greedy-cost heuristic sorts all the inter-AS traffic flows in descending order of bandwidth demand and selects one at a time in that order. From the bandwidth offers that have sufficient bandwidth to accommodate the given traffic flow, we select the one which incurs the least IBP cost. The flow is then allocated to this bandwidth offer and its corresponding inter-AS route. This step is repeated for the next traffic flow until all flows have been considered. One can imagine this heuristic might be a conventional algorithm used by INPs to solve the IBP problem.

Greedy-random heuristic: A greedy-random heuristic algorithm is included as a baseline comparison. The random heuristic algorithm is similar to the greedy-cost heuristic except that the bandwidth offer selection of traffic flows is done at random. It may be viewed as the solution obtained by a trial-and-error or an ad hoc IBP approach.

7.5.1. Evaluation of the total IBP cost

The aim of the proposed GA is to achieve better and near-optimal IBP cost in comparison with the alternative algorithms. Hence, the main objective of the evaluation in this section is to quantify the effectiveness of the proposed GA over the alternative algorithms.

Fig. 7 shows the total IBP cost achieved by the greedy-cost and the GA as a function of inter-AS traffic flows. The performance of the greedy-random heuristic is not presented in this figure since it has a significant performance gap from the other heuristics. Nevertheless, it is compared to the alternative algorithms in Table 3. The legend in the figure shows the names of the algorithms followed by the percentage of established peering connections as mentioned at the beginning of Section 5.

The figure presents the results of two practical scenarios, and we evaluate whether the proposed GA performs consistently well under these scenarios. The first scenario consists of all customer–provider connections. In other

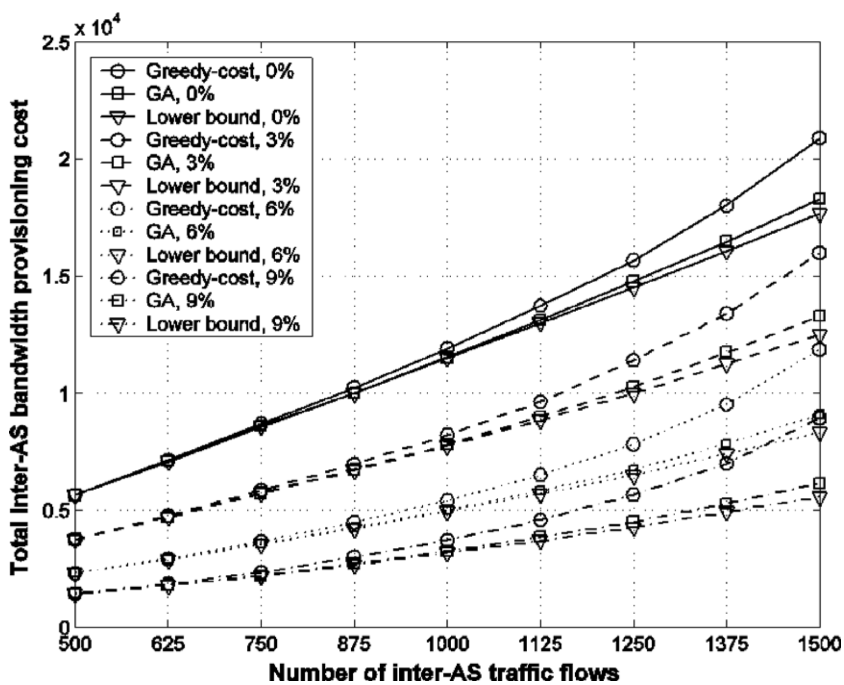


Fig. 7. Evaluation of the total inter-AS bandwidth provisioning cost.

Table 3
Performance improvement of the GA over the alternative algorithms (%)

Number of Inter-AS traffic flows	1000	1125	1250	1375	1500
Over greedy-cost with 0% peering	3.33	5.0	5.92	8.67	12.75
Over random with 0% peering	76.16	75.97	75.68	75.6	75
Over greedy-cost with 3% peering	4.98	6.91	10.13	12.61	17.16
Over random with 3% peering	83.66	83.08	83.06	81.95	81.38
Over greedy-cost with 6% peering	7.71	10.6	14.3	18.01	24.0
Over random with 6% peering	89.22	88.7	88.47	87.67	87
Over greedy-cost with 9% peering	12.59	16.45	20.96	24.87	31.76
Over random with 9% peering	92.7	92.41	91.98	91.47	90.85

words, no peering connection (i.e. 0%) is established and the charge of each bandwidth offer is non-zero. We generate an integer uniformly between 1 and 10 to represent each cost. The figure shows that the GA has a lower total IBP cost at all numbers of inter-AS traffic flows. We conjecture that when the number of inter-AS traffic flows is small, the inter-AS links and the bandwidth offers have relatively plenty of bandwidth to cover all the traffic, and so the GA and the greedy-cost algorithm would give equivalent IBP results and costs. In contrast, as the number of inter-AS traffic flows increases, both the overall inter-AS link and bandwidth offer utilizations increase and some inter-AS links or bandwidth offers have even reached their capacity limits. In this case, some traffic flows may be assigned to other bandwidth offers which have higher costs. This evaluation shows that a careful selection of bandwidth offers is important in order to minimize the total IBP cost. This can be achieved by the GA.

In addition, the total IBP costs of the GA at all volumes of traffic flows are closer to the lower bound than the greedy-cost heuristic. This shows that the GA is not only able to achieve a better cost than the greedy-cost, but also able to achieve a near-optimal cost.

In the second scenario not only are customer–provider connections considered but also peering connections. We evaluate three levels of established peering connections: 3%, 6% and 9% of the total number of bandwidth offers. Simulation data presented in this scenario is as for the previous one except that a designated number of bandwidth offers is randomly selected as peering connections. In current Internet peering practice, most ASes will only accept on a peer link traffic from the peers' customers. Since our purpose is to merely evaluate the performance of the algorithms, we follow the assumption in [39] that general policy routing and peering/transit restrictions are ignored.

Fig. 7 shows that the GA always performs better than the greedy-cost at all degrees of peering connection and all number of inter-AS traffic flows. This is similar to the results of the 0% peering scenario. The GA has better total IBP costs than the greedy-cost heuristic as the degree of peering connection increases. This is because more and more peering connections do not incur any charges, so that the GA can more effectively utilize the cost-free bandwidth in order to further minimize the total IBP cost. In general, this performance improvement not only applies to the second scenario where some peering connections exist but also applies to the 0% peering scenario where some exceptional low cost bandwidth offers exist.

Table 3 shows the relative improvement of the GA over the greedy-cost and the greedy-random heuristic algorithms at all number of inter-AS traffic flows with different degrees of peering connection. By summarizing the table and considering a reasonably high traffic volume, we find that the proposed GA has approximately 5–30% and 75–90% performance improvement over the greedy-cost and the greedy-random heuristics, respectively, under different scenarios. In comparison with the greedy-random heuristic, the performance of the GA is remarkable. This shows the importance and value of using systematic approaches, such as the proposed GA, over the trial-and-error and ad hoc approaches.

7.5.2. Evaluation of the proposed GA average running time

In Table 4 we provide the average running time of the GA. The average running time increases as the number of traffic flows increases. We can see that even for quite high numbers of traffic flows the running times are acceptable. These times are perfectly acceptable taking into account the timescale of the provisioning system operation.

7.5.3. Evaluation of cost functions

In Section 5.5, we assumed that the cost associated with bandwidth provisioning is constant, i.e. a linear cost function. However, such linear cost functions sometimes may not reflect actual operations such as economies of scale in which the cost per unit bandwidth decreases as the amount of bandwidth purchased increases. In this section, we investigate the IBP problem using a concave cost function.

Referring to Larsson et al. [40] and LeBlanc [41], we assume the concave cost function to be $\theta(\sigma)^{\alpha}$ where θ is the per bandwidth unit cost, σ is the amount of bandwidth

Table 4
Average running time of the GA

Number of traffic flows	Average running time (s)
500	36.6
1000	78.6
1500	150.4

procured and α is the degree of concavity. We replace (1) of the IBP problem with the concave cost function:

$$\text{Minimize} \sum_{k\in K}\sum_{j\in J}\sum_{n\in NEXT_j} Chg_k^{j,n}\cdot\left(\sum_{i\in I} x_{i,k}^{j,n}\cdot t'(i,k)\right)^{\alpha} \quad (29)$$

The other problem constraints (2)–(7) remain unchanged.

To cope with the concave cost function, the IBP algorithms presented have to be slightly modified as follows:

For the GA, the fitness value (step 3) of a chromosome is evaluated by objective function (29). In step 7, the two heuristic operators are changed to consider the lowest marginal inter-AS bandwidth provisioning cost for each traffic flow assignment. The marginal cost is the additional cost of purchasing more units of bandwidth over the amount of bandwidth that has already purchased.

For the greedy-cost and greedy-random heuristics, the criteria of outbound provider SLA selection are changed to consider the lowest marginal IBP cost for each traffic flow assignment.

We evaluate the modified IBP algorithms with a concave cost function with four different concavities α (Fig. 8): 0.8, 0.85, 0.9 and 0.95.

Fig. 9 shows the total IBP cost achieved by the modified GA and the modified greedy-cost heuristic as a function of inter-AS traffic flows. The GA has lower IBP costs at all values of α. As the value of α decreases, the total IBP cost decreases because of large economies of scale, and the relative improvement of the GA over the greedy-cost heuristic increases. The reason for this performance improvement is similar to the one that explained the increasing performance improvement for decreasing peering connection degrees. The similarity is that both cases have some attractive factors (respectively, very low cost and large economies of scale) that contribute to the remarkable performance achieved by the GA.

7.5.4. Discussion of the IBP algorithms

The simulation study in this section has evaluated the performance of three IBP algorithms. Simulation results have firstly shown that the proposed GA is efficient and is able to achieve better total IBP cost than the random-based and the conventional heuristic algorithms. The relative total IBP cost improvement achieved by the GA over the greedy-cost heuristic and the random-based algorithms are great, with 5–30% and 75–90% cost savings, respectively. In addition, the GA performs consistently well with different concave cost functions. We conclude that the IBP solutions obtained by the proposed GA are good overall. This has an implication for INPs that a systematic approach could be developed to optimize the total IBP cost significantly.

7.6. Evaluation of the TA algorithms

The previous section evaluated the performance of the proposed IBP algorithms. Once the IBP phase is completed, an AS performs TA to optimize network resource utilization in order to provide end-to-end bandwidth guarantees for the supported traffic. In this section, we evaluate the performance of our proposed TA algorithms.

We assume that outbound provider SLAs are successfully established in line with the first scenario in the evaluation of IBP algorithms, i.e. the GA IBP outcomes with a linear cost function and all customer–provider connections (0% peering). These outbound provider SLAs are then the

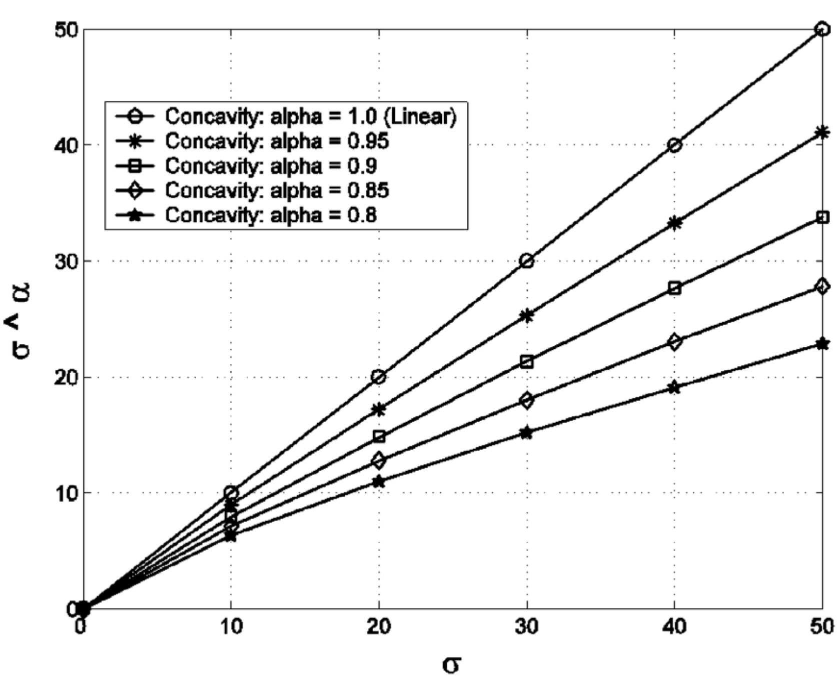


Fig. 8. Concave cost functions.

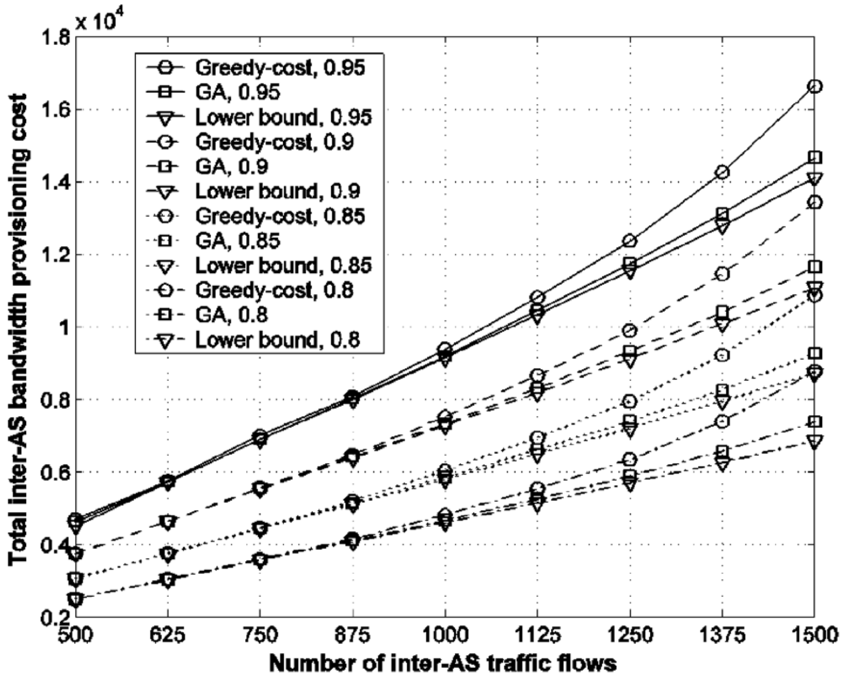


Fig. 9. Evaluation of the total inter-AS bandwidth provisioning cost with different concave cost functions.

input to the TA problem. We consider the following three approaches for the TA problem, namely Cost-only, Cost–Performance and Performance-only approaches. The words "Cost" and "Performance" used in the names of these approaches mean that the ordered priorities of the algorithm optimization targets are on the total IBP cost and the total bandwidth consumption, respectively.

Cost-only: Given an IBP solution produced by the GA, there are multiple solutions for assigning traffic to satisfy all the TA problem constraints. Any of these solutions can be selected as the solution of the Cost-only approach since it does not optimize the total bandwidth consumption in the network. We use the Random-TA heuristic algorithm (Fig. 10) to find a solution for the Cost-only approach.

Cost–Performance: Given an IBP solution produced by the GA, the Cost–Performance approach takes the proposed greedy-penalty heuristic algorithm as the TA algorithm to optimize the total bandwidth consumption in the network.

Performance-only: The Performance-only approach does not use the IBP solution. Instead, it takes all the bandwidth offers (rather than the outbound provider SLAs) as input and uses the greedy-penalty heuristic algorithm to solve the TA problem. The total IBP cost is then equal to the sum of the cost of each accepted bandwidth offer. Since the total IBP cost is calculated by taking overprovisioning into consideration, we approximate the total IBP cost of the Performance-only approach by multiplying its solution cost by f_{over} in order to compare it with the total IBP costs achieved by the other two approaches.

7.6.1. Cost vs. performance

In this section we evaluate the proposed three TA approaches. We test the hypothesis that the greedy-penalty heuristic algorithm can improve the total network bandwidth consumption.

Fig. 11 shows the total IBP costs of all the TA approaches at three different volumes of inter-AS traffic flows: 500, 1000 and 1500. The total IBP costs are normalized by the cost of the solution produced by the GA. The total IBP costs of the Cost-only and the Cost-Performance approaches are identical because they both use the IBP solution produced by the GA. In contrast, the total IBP cost of the Performance-only approach is on average 4 times higher than the others. This significantly higher cost

Random-TA Heuristic Algorithm

Sort inter-AS traffic flows in decreasing order of bandwidth demand
For each traffic flow in that order
- Assign an egress point randomly to the traffic flow
- Establish a bandwidth constrained path between the ingress and egress point
- Update utilized resources

End For

Fig. 10. The Random-TA heuristic.

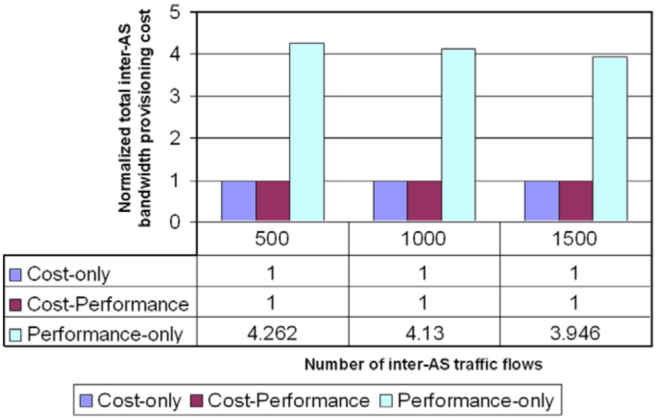


Fig. 11. Normalized total inter-AS bandwidth provisioning cost.

results from neglecting the IBP optimization so that some expensive bandwidth offers are selected, although, as we can see below, using them can significantly improve the total bandwidth consumption in the network.

Indeed, although the Performance-only approach has a very high total IBP cost, Fig. 12 shows that its total bandwidth consumption is approximately half of the other two approaches. Nevertheless, because of its high total IBP cost, the Performance-only approach can be assumed impractical. This implies that there can be conflict between the IBP cost and bandwidth consumption. Therefore, we need a compromising solution that would balance the interests of these two metrics. The Cost–Performance approach attempt to achieve such solution as it has low IBP cost and low total bandwidth consumption compared to the Cost-only approach with the amount closer to the Performance-only approach. This reduced total bandwidth consumption reveals that the proposed greedy-penalty heuristic algorithm has on average a 10% improvement over the Random-TA heuristic algorithm.

7.6.2. Evaluation of the greedy-penalty heuristic algorithm average running time

Table 5 provides the average running time of the proposed greedy-penalty heuristic algorithm. The average running time increases as the number of traffic flows increases. These running times are perfectly acceptable taking into account the timescale of the provisioning system operation. The computation time could have been much longer if GA was used due to its evolutionary process.

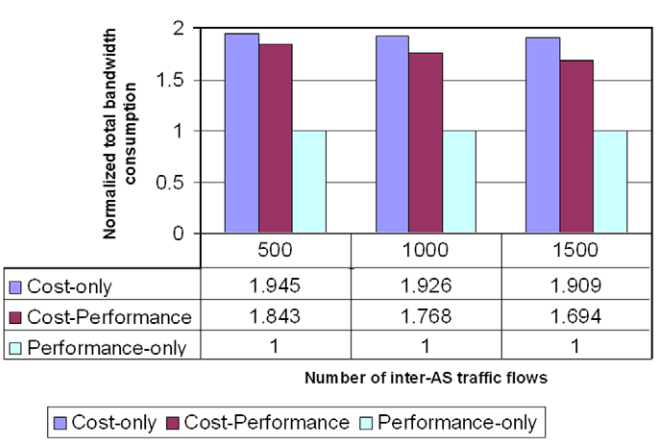


Fig. 12. Normalized total bandwidth consumption in the network.

Table 5
Average running time of the greedy-penalty heuristic

Number of traffic flows	Average running time (s)
500	6.2
1000	22.1
1500	64.76

7.6.3. Discussion of the TA approaches

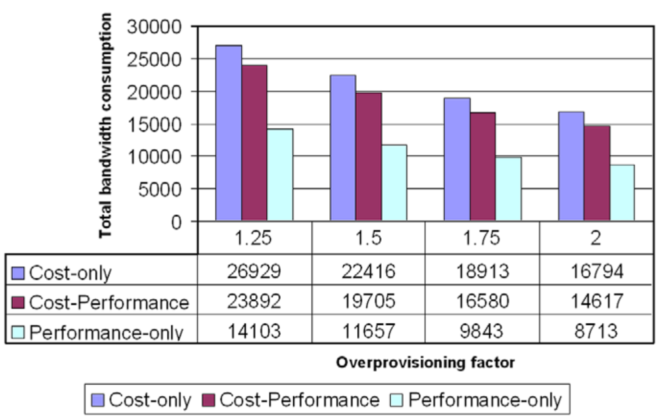
The simulation described in this section has evaluated the performance of three TA approaches. Simulation results have shown that the proposed greedy-penalty heuristic algorithm used by the Cost–Performance approach is efficient and is able to achieve on average 10% less total bandwidth consumption than the random-based algorithm used in the Cost-only approach. The performance difference between the Performance-only approach and the other two reveals that a trade-off exists between the IBP and the TA optimization. This trade-off has also been discussed in [19] where primarily optimizing monetary cost can degrade network performance and vice versa. However, the determination of relative weights between cost and performance optimizations is far from trivial, particularly when the units of the two metrics have different scales. It is thus in many cases difficult to express in terms of weights the trade-off between the two metrics. Therefore, we assume that from business point of view, an AS considers the IBP cost optimization as more important than the TA performance optimization. Based on this assumption and the simulation study in this section, we conclude that the Cost–Performance approach, which uses our proposed GA and the greedy-penalty heuristic algorithm, performs well both in terms of the total IBP cost and the total bandwidth consumption, in comparison with the Cost-only and the Performance-only approaches.

The Cost–Performance approach can be used by INPs to achieve an effective provisioning of end-to-end bandwidth guarantees. Moreover, since the TA problem has dealt with the selection of inter-AS route and explicit intra-AS route within the backbone network, the Cost–Performance approach could be effectively applied to BGP/MPLS virtual private network provisioning [42], a subject which is currently attracting a great deal of attention.

7.6.4. Impact of inter-AS provisioning factor

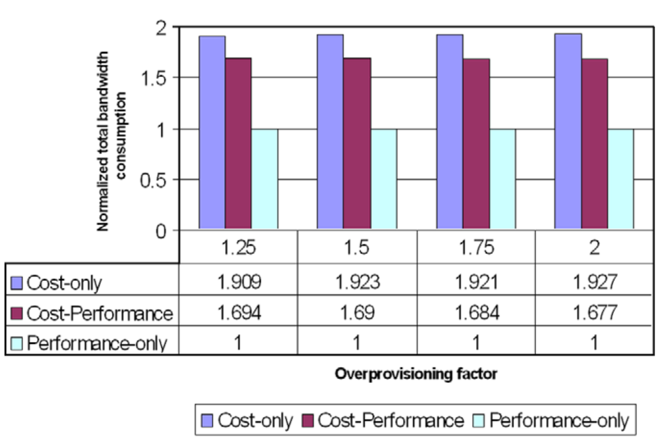
In this section, we evaluate the impact of overprovisioning factor on the total bandwidth consumption achieved by the three TA approaches. The results of this evaluation are based on 1500 inter-AS traffic flows. The values of the inter-AS bandwidth overprovisioning factor examined are 1.25, 1.5, 1.75 and 2.0. As the inter-AS available bandwidth increases, the outbound provider SLA capacity constraint becomes less restrictive to the TA problem. Thus, in this case, we expect that the total bandwidth consumption in the network can be further improved.

Fig. 13 shows that the total bandwidth consumption decreases as the overprovisioning factor increases. This is because a large overprovisioning factor reduces the outbound provider SLA capacity constraint and therefore increases the solution space for the TA algorithm, enabling it to find a result with lower total bandwidth consumption. As expected, the Cost–Performance approach has lower total bandwidth consumption than the Cost-only approach. Fig. 14 shows the normalized total bandwidth consumption achieved by the three TA approaches. As the overprovisioning factor increases, the relative improvement of the Cost–Performance approach over the Cost-only approach slightly increases from approximately 11% to 13%. Fig. 14 also reveals that the performance differences among the three TA approaches are consistent and are insensitive to changes on the overprovisioning factor.



	1.25	1.5	1.75	2
Cost-only	26929	22416	18913	16794
Cost-Performance	23892	19705	16580	14617
Performance-only	14103	11657	9843	8713

Fig. 13. Total bandwidth consumption achieved by different f_{over}.



	1.25	1.5	1.75	2
Cost-only	1.909	1.923	1.921	1.927
Cost-Performance	1.694	1.69	1.684	1.677
Performance-only	1	1	1	1

Fig. 14. Normalized total bandwidth consumption achieved by different f_{over}.

The results presented in the figures have revealed the effect of IBP on the TA performance with a different overprovisioning factor. The results confirm our conjecture that as the overprovisioning factor increases, more bandwidth is available in outbound provider SLAs for the TA algorithms to further optimize the total bandwidth consumption.

8. Conclusion

In this paper we have reviewed a cascaded negotiation model for negotiating and establishing SLAs for bandwidth guarantees between ASes, and a network dimensioning system to solve the inter-AS bandwidth provisioning and the traffic assignment problems systematically.

We formulated the inter-AS bandwidth provisioning problem as an integer-programming problem and showed it to be NP-hard. An efficient genetic algorithm was proposed to solve the problem. Our simulation study shows that the genetic algorithm has a near-optimal total inter-AS bandwidth provisioning cost. This cost is approximately 5–30% and 75–90% less than the cost achieved by a conventional greedy heuristic algorithm and a random-based algorithm, respectively, under two customer-peering scenarios. In addition, the genetic algorithm performs consistently well under different concave cost functions.

We formulated the traffic assignment problem as an integer-programming problem and showed it to be NP-hard. An efficient greedy-penalty heuristic algorithm was proposed to solve the problem. Our simulation study showed that the greedy-penalty heuristic algorithm achieved on average 10% less total bandwidth consumption than the random-based TA heuristic algorithm.

Finally, we evaluated the effects of different overprovisioning factor values on the total bandwidth consumption. The more the inter-AS bandwidth is overprovisioned, the less the total bandwidth is needed to carry the supported traffic across the network.

A limitation of our work is performance robustness. In case where the derived traffic matrix deviates significantly from the real traffic demands or link failures happen, the performance of IBP and TA may be affected since these network conditions have not been taken into account during the optimization. As future work, we will make the IBP and TA problems robust to traffic demand uncertainty and link failures. Although this may result in trade-offs between performance and robustness, we attempt to achieve good and balance solutions with respect to these two metrics.

Acknowledgements

This work was undertaken in the context of the Information Society Technologies FP5 Management of End-to-end Quality of Service Across the Internet at Large

(MESCAL-IST-2001-37961) and FP6 AGAVE (IST-027609) research projects, which are partially funded by the Commission of the European Union.

References

[1] R. Zhang et al., MPLS inter-AS traffic engineering requirements, IETF Internet Draft, draft-ietf-tweg-interas-mpls-te-req-09.txt, September, 2004.
[2] D. Awduche et al., Overview and principles of internet traffic engineering, IETF RFC 3272 (2002).
[3] The European Union IST TEQUILA project. Available from: <www.ist-tequila.org>.
[4] P. Trimintzios et al., Service-driven traffic engineering for intra-domain quality of service management, IEEE Network, Special Issue on Network Management of Multi-service, Multimedia, IP-based Networks 17 (3) (2003) 29–36.
[5] P. Trimintzios et al., Quality of service provisioning through traffic engineering with applicability to IP-based production networks, computer communications, Special Issue on Performance Evaluation of IP Networks and Services 26 (8) (2003) 845–860.
[6] I. Akyildiz et al., A new traffic engineering manager for DiffServ/MPLS networks: design and implementation on an IP QoS testbed, Computer Communications 26 (4) (2003) 388–403.
[7] S. Srivastava et al., Benefits of traffic engineering using QoS routing schemes and network controls, Computer Communications 27 (5) (2004) 387–399.
[8] B. Quoitin et al., Interdomain traffic engineering with BGP, IEEE Communications Magazine (2003) 122–128.
[9] The European Union IST MESCAL project. Available from: <www.mescal.org>.
[10] M. Howarth et al., Provisioning for inter-domain quality of service: the MESCAL Approach, IEEE Communications Magazine 43 (6) (2005) 129–137.
[11] S. Bakiras, V.O.K. Li, A scalable architecture for end-to-end QoS provisioning, Computer Communications 27 (13) (2004) 1330–1340.
[12] K. Nichols, V. Jacobson, L. Zhang, A two-bit differentiated services architecture for the internet, IETF RFC 2638 (1999).
[13] C. Fraleigh, F. Tobagi, C. Diot, Provisioning IP backbone networks to support latency sensitive traffic, in: Proceedings of IEEE INFOCOM 2003, April, 2003, pp. 375–385.
[14] Z. Duan, Z.L. Zhang, T.Y.W. Hou, Service overlay networks: SLAs, QoS and bandwidth provisioning, IEEE/ACM Transactions on Networking 11 (6) (2003) 870–883.
[15] N. Feamster et al., Guidelines for interdomain traffic engineering, ACM SIGCOMM Computer Communications Review (2003) 19–30.
[16] T.C. Bressoud, R. Rastogi, M.A. Smith, Optimal configuration for BGP route selection, in: Proceedings of IEEE INFOCOM, 2003, pp. 916–926.
[17] A. Akella, A. Shaikh, R. Sitaraman, A measurement-based analysis of multihoming, in: Proceedings of ACM SIGCOMM Conference 2003, August, 2003, pp. 353–364.
[18] A. Akella et al., A comparison of overlay routing and multihoming route control, in: Proceedings of ACM SIGCOMM Conference 2004, August, 2004, pp. 93–106.
[19] D.K. Goldenberg et al., Optimizing cost and performance for multihoming, in: Proceedings of ACM SIGCOMM Conference 2004, August, 2004, pp. 79–92.
[20] N. Feamster et al., A model of BGP routing for network engineering, in: Proceedings of ACM SIGMETRICS Conference, June, 2004, pp. 331–342.
[21] A. Feldmann et al., Deriving traffic demands for operational IP networks: methodology and experience, IEEE/ACM Transactions on Networking 9 (3) (2001) 265–279.
[22] N. Duffield, C. Lund, M. Thorup, Estimating flow distributions from sampled flow statistics, in: Proceedings of ACM SIGCOMM Conference 2003, August, 2003, pp. 325–336.
[23] R. Guerin, H. Ahmadi, M. Naghshineh, Equivalent capacity and its applications in high-speed networks, IEEE Journal on Selected Areas in Communications 9 (7) (1991) 968–981.
[24] R. Teixeira, N. Duffield, J. Rexford, M. Roughan, Traffic matrix reloaded: impact of routing changes, in: Proceeding of Passive and Active Measurement Workshop, March/April, 2005, pp. 251–264.
[25] A. Feldmann et al., A methodology for estimating interdomain web traffic demand, in: Proceedings of ACM SIGCOMM Internet Measurement Conference 2004, October, 2004, pp. 322–335.
[26] A. Nucci et al., Increasing link utilization in IP over WDM networks using availability as QoS, Photonic Network Communication Journal, Special Issue on Mesh Restoration 9 (1) (2005) 55–75.
[27] B. Gavish, H. Pirkul, Algorithms for the multi-resource generalized assignment problem, Management Science 32 (1991) 695–713.
[28] J.H. Holland, Adaptation in Natural and Artificial System, The University of Michigan Press, 1975.
[29] P.C. Chu, J.E. Beasley, A genetic algorithm for the generalized assignment problem, Computers and Operations Research 24 (1) (1997) 17–23.
[30] S. Martello, P. Toth, Knapsack Problems: Algorithms and Computer Implementations, Wiley Interscience, 1990.
[31] K.H. Ho et al., Multi-objective egress router selection policies for inter-domain traffic with bandwidth guarantees, in: Proceedings of IFIP Networking 2004, May, 2004, pp. 271–283.
[32] M. Pioro, D. Medhi, Routing, Flow and Capacity Design in Communication and Computer Networks, Morgan Kaufmann Series in Networking, 2004.
[33] E. Osborne, A. Simha, Traffic Engineering with MPLS, Cisco Press, 2002.
[34] T.H. Cormen, C.E. Leiserson, R.L. Rivest, Introduction to Algorithms, MIT Press, Cambridge, MA, 1990.
[35] BRITE: Boston University Representative Internet Topology Generator. Available from: <http://www.cs.bu.edu/brite/>.
[36] W. Fang, L. Peterson, Inter-AS traffic patterns and their implications, in: Proceedings of IEEE GLOBECOM 1999, December, 1999, pp. 1859–1868.
[37] A. Broido et al., Their shares: diversity and disparity in IP traffic, in: Proceedings of Passive and Active Measurement Workshop, 2004, pp. 113–125.
[38] 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.
[39] J. Feigenbaum et al., A BGP-based mechanism for lowest-cost routing, in: Proceedings of ACM Symposium on Principles of Distributed Computing, July, 2002, pp. 173–182.
[40] T. Larsson, A. Migdalas, M. Ronnqvist, A Lagrangian heuristic for the capacitated concave minimum cost network flow problem, European Journal of Operational Research 78 (1994) 116–129.
[41] L.J. LeBlanc, Global solutions for a nonconvex, nonconcave rail network model, Management Science 23 (1976) 131–139.
[42] E. Rosen, Y. Rekhter, BGP/MPLS VPNs, IETF RFC 2547 (1999).
[43] K.H. Ho et al., On egress router selection for inter-domain traffic with bandwidth guarantees, in: Proceedings of IEEE Workshop on High Performance Switching and Routing (HPSR) 2004, April, 2004, pp. 337–342.
[44] K.H. Ho et al., An incentive-based quality of service aware algorithm for offline inter-AS traffic engineering, in: Proceedings of IEEE Workshop on IP Operations and Management (IPOM) 2004, October, 2004, pp. 34–40.
[45] M.R. Garey, D.S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, Freeman, San Francisco, 1979.

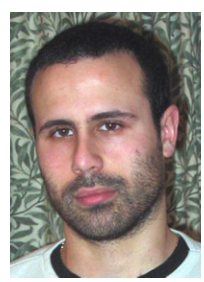
[46] D. Mitra, K.G. Ramakrishnan, A case study of multiservice, multipriority traffic engineering design for data networks, in: Proceedings of IEEE GLOBECOM 1999, pp. 1077–1083.
[47] D.E. Goldberg, Genetic Algorithm in Search, Optimization, and Machine Learning, Addison-Wesley, Reading, MA, 1989.
[48] P. Pan, E.L. Hahne, H. Schulzrinne, BGRP: sink-tree-based aggregation for inter-domain reservations, Journal of Communications and Networks 2 (2) (2000) 157–167.
[49] P. Sampatakos et al., BGRP: quiet grafting mechanisms for providing a scalable end-to-end QoS solution, Computer Communications 27 (2004) 423–433.
[50] R. Sofia, R. Guerin, P.Veiga, Enabling scalable inter-AS signaling: a load reduction approach, in: Proceedings of IEEE ISCC, 2005, pp. 827–832.

Kin-Hon Ho is a research fellow at the Centre for Communication Systems Research (CCSR), University of Surrey, United Kingdom. He holds a Bachelor's degree in computer studies from City University of Hong Kong, and an M.Sc. (Eng.) in Data Communications from the University of Sheffield, and a Ph.D. from the University of Surrey, United Kingdom. His research interests include Internet traffic engineering and network optimization.

Michael Howarth is a lecturer in networking at the CCSR, University of Surrey, United Kingdom. He holds a Bachelor's degree in engineering science and a D.Phil. in electrical engineering, both from Oxford University, and an M.Sc. in telecommunications from the University of Surrey. His research interests include IP multicast, Internet traffic engineering, security systems, protocol design and optimization, and satellite communications.

Ning Wang is a research fellow at CCSR, University of Surrey, United Kingdom. He holds a B.Eng. in computer science from Changchun University of Science and Technology, PR China, an M.E. in electrical and computer engineering from Nanyang Technological University, Singapore, and a Ph.D. in electronic engineering from the University of Surrey. His research interests include multicast communication, QoS provisioning, and Internet traffic engineering.

George Pavlou is a Professor of Communication and Information Systems at the CCSR, University of Surrey, United Kingdom, where he leads the activities of the Networks Research Group. He holds a BEng in Electrical and Mechanical Engineering from the National Technical University of Athens, Greece, and MSc and Ph.D. degrees in Computer Science from University College London, UK. He is responsible for a number of European and UK research projects and industrial collaborations. His research interests include network planning and dimensioning, traffic engineering, multicast, mobile ad hoc networks, multimedia service control, programmable networks and communications middleware. He has published about 100 papers in refereed international conferences and journals and has contributed to standardization activities in ISO, ITU-T, TMF, OMG and IETF.



Stylianos Georgoulas was admitted to the Department of Electrical and Computer Engineering at the University of Patras in Greece and received his Diploma in 2001, having followed the Telecommunications and Information Technology pathway. He is currently a Research Student at the CCSR, University of Surrey, where he is also a Ph.D. candidate. While at Surrey he has been involved in the IST TEQUILA and EU MESCAL projects. His research interests are in the area of admission control and bandwidth management in IP differentiated services networks.