An Overview of Routing Optimization for Internet Traffic Engineering

Ning Wang, Kin Hon Ho, George Pavlou and Michael Howarth

Abstract

Traffic Engineering (TE) is an important mechanism for Internet Network Providers (INPs) seeking to optimize network performance and traffic delivery. Routing optimization plays a key role in traffic engineering, finding efficient routes so as to achieve the desired network performance. In this article, we review Internet traffic engineering from the perspective of routing optimization. We provide a taxonomy of routing algorithms in the literature, dating from the advent of the TE concept in the late 1990s. We classify the algorithms into multiple dimensions, namely unicast/multicast, intra-/inter-domain, IP-/MPLS-oriented and offline/online TE schemes. In addition, we investigate some important traffic engineering issues, including robustness, TE interactions and interoperability with overlay selfish routing. While revisiting the existing solutions, we also point out some important issues that are worthy of investigation in future research activities.

1. Introduction

The Internet is currently experiencing a transition from point-to-point Best Effort (BE) communications towards a multi-service network that supports many types of multimedia applications, with potentially high bandwidth demand. Thanks to the rapid development of communication network hardware, adding physical resources (e.g., fast-speed switching and routing elements, high capacity network links etc.) to the existing Internet has become relatively cheap in recent years. Typically, the advent of increasingly high-speed links has offered opportunities for IP Network Providers (INPs) to adopt a strategy of bandwidth over-provisioning in their networks. Nevertheless, this approach is currently only applicable to the core network, and the demand from sharply growing customer traffic over the global Internet still cannot be satisfied. The measurement results presented in [1] indicate that bottlenecks of the Internet backbone are not only located at inter-domain links between Autonomous Systems (ASes), but also within individual domains. Given this information, it is essential for INPs to perform efficient resource optimization both intra- and inter-domain so as to eliminate these bottlenecks. Internet Traffic Engineering (TE) is the process of performing this task. In [2], TE is defined as large-scale network engineering for dealing with IP network performance evaluation and optimization. A more straightforward explanation of TE is also given in [3]: "to put the traffic where the network bandwidth is available". From this statement, we note that the fundamental task of traffic engineering is to perform appropriate route selection such that the given bandwidth capacity is able to support maximum customer traffic without causing network congestion. From this perspective, the nature of traffic engineering is effectively a routing optimization for enhancing network service capability. Figure 1 illustrates this with a simple TE example. We assume that the bandwidth capacity of each link is 10Mbps, and there are three individual customer flows injected at node A, heading towards node C. If conventional shortest path routing is applied, all the customer flows are routed on the direct link A-C, thus causing the link utilization to be as high as 180% ($6\times3/10$). On the other hand, if the three flows are routed through different paths, as shown in Figure 1(b), the total traffic within the network is evenly distributed without causing link congestion. As this example illustrates, routing optimization that uses alternative multiple paths other than conventional shortest path based approaches can be an effective means to improving the network service capability.

Two major issues that have recently received attention in TE approaches are Quality of Service (QoS) and resilience. First, many of the new multimedia applications not only have bandwidth requirements, but also require other QoS guarantees, such as end-to-end delay, jitter or packet loss probability. These QoS requirements impose new challenges on INPs' traffic engineering in that the end-to-end QoS demands need to be satisfied through TE mechanisms. Second, given the fact that network node and link failure are still frequent events on the Internet, TE solutions have to consider how to minimize the impact of failures on the network performance and resource utilization. There exists a large amount of work in the literature on QoS routing and path protection/restoration respectively. In order to restrict the scope of our survey, it is worth clarifying the relationship and difference between TE and QoS routing / resilience schemes. According to [4], TE objectives can be classified into traffic-oriented and resource-oriented. Most QoS-aware and resilience-aware TE schemes belong to the traffic-oriented category, which puts more emphasis on improving the performance perceived by the customer sending traffic. According to this criterion, if a QoS routing scheme is implemented exclusively from a customer's viewpoint without considering global network optimization, then it is known as *selfish routing*; we do not consider this in this article, although we note that a comprehensive survey on QoS aware selfish routing can be found in [5]. As far as resilience is concerned, the objective is to avoid sub-optimal resource utilization (resource oriented) and negative impacts on traffic delivery (traffic oriented) in case of link/node failure. We will discuss detailed robustness-aware TE solutions in section 6.

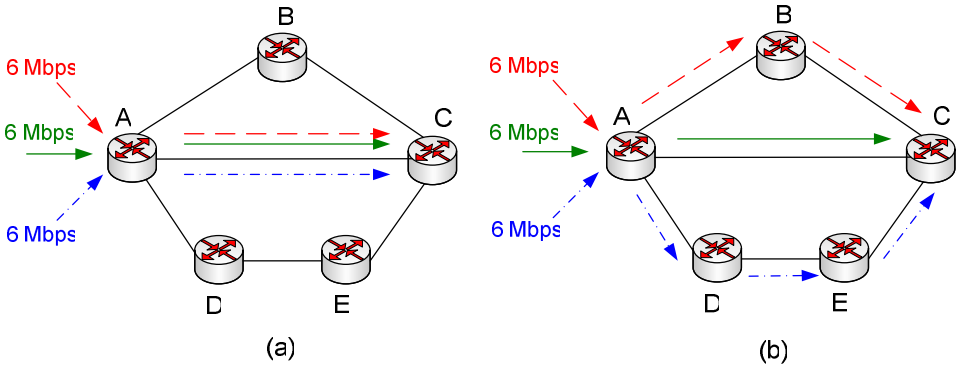

Figure 1. A simple TE example

Many papers have been published in the area of routing optimization. As a result, it is by no means an easy task to classify various TE solutions, and present a comprehensive and clear survey. In this paper, we classify these TE routing approaches according to four orthogonal criteria: (1) traffic optimization scope, (2) routing enforcement mechanism, (3) time/state dependence or availability of traffic demand and (4) traffic type. First of all, TE can be classified into two categories according to the scope: *intra-domain* and *inter-domain*. In intra-domain TE, optimization focuses on how to control traffic routing within a single AS. In contrast, inter-domain TE considers how to optimize traffic that travels across multiple ASes. Inter-domain TE paradigms can be generally classified into two categories. The first, which has been extensively addressed is how to control inter-domain traffic within the local AS, e.g., to find optimal ingress/egress points for inter-domain traffic that is injected into or delivered out of the local domain. The second category, which has not yet been well studied, considers "end-to-end" TE optimization across multiple ASes. In this scenario, individual ASes may need cooperation with each other in order to deliver the traffic over the desired inter-domain routes. Second, from the perspective of routing enforcement, there exist two distinct TE mechanisms, *IP-based* and *MPLS-based*. For IP-based TE, routing is optimized by adjusting the routing parameters of the underlying IP routing protocols such as OSPF/ISIS and BGP.

On the other hand, MPLS-based TE adopts packet encapsulation and explicit routing with dedicated Label Switching Paths (LSPs). Third, traffic engineering can be categorized into *offline* and *online*. In offline TE, all traffic demands from customers are assumed to be known *a priori* to some greater or lesser extent, and the TE task is then tp efficiently map the predicted traffic demand onto the physical network. In contrast, for the online TE case, the INP needs to perform lightweight and efficient path selection one by one for each incoming flow, without knowing any traffic demand in advance. Finally, we should mention that Internet traffic consists of different types of flows, such as IP unicast, multicast and various types of overlay traffic such as Virtual Private Network (VPN) and Content Distribution Network (CDN) flows. Routing optimization of these different traffic types may require different solutions. In this paper we will survey not only the common *unicast TE*, but also *multicast TE* which is emerging as a popular approach given recent progress in IP multicast.

To summarize, an overall taxonomy of Internet traffic engineering is presented in Figure 2, and this article is organized following the structure of this diagram. The objective of this article is thus to provide a comprehensive survey on routing optimization for all the components in the TE hierarchy. The rest of the article is organized as follows. We specify in Section 2 the detailed characteristics of different types of TE according to Figure 2. In Section 3 we introduce intra-domain traffic engineering, which includes both MPLS and IP-based routing optimization algorithms. In Section 4 we move on to inter-domain traffic engineering, which we further divide into inbound and outbound TE. In Section 5, multicast traffic engineering is presented. We then discuss in Section 6 some important interactions between current traffic engineering approaches. Finally we provide a summary in Section 7. It is worth mentioning that this survey does not claim to be exhaustive, although we attempt not to omit any important works in the area.

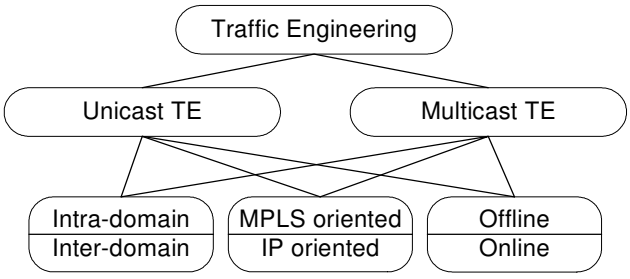

Figure 2. Hierarchical classification of Internet traffic engineering

2. Traffic Engineering Classifications

2.1 Intra-domain TE vs. Inter-domain TE

The task of intra-domain traffic engineering is to optimize customer traffic routing between AS border routers (ASBRs) within a single domain. In comparison, inter-domain traffic engineering deals with the problem of optimizing inter-domain traffic traveling across multiple ASes. As mentioned above, most of the existing literature focuses on how to select ASBRs optimally as the ingress/egress points for inter-domain traffic that travels across the local AS. That is to say, if the traffic has multiple potential ASBRs from which it can enter or leave the local domain, then the problem of inter-domain TE for an INP is: “which ASBR(s) should be used as the ingress/egress point(s) for routing the traffic through the local network,

so that the network resource utilization is optimized?" According to the control over how traffic enters/leaves the domain, inter-domain traffic engineering can be further classified into inbound TE and outbound TE. Figure 3 presents a simple example to illustrate the difference between intra- and inter-domain traffic engineering semantics, specifically using outbound traffic engineering as an example for inter-domain TE. We assume that traffic destined to the remote prefix 20.20.20.0/24 (AS200) is injected into the local AS (AS100, 10.10.10.0/24) via ASBR 10.10.10.3, and both the internal peers 10.10.10.1 and 10.10.10.2 can provide a route to AS200 (i.e., both routers receive reachability information towards 20.20.20.0/24 through external BGP advertisements). In this scenario, the decision to use ASBR 10.10.10.1 or 10.10.10.2 (or both for load balancing with inter-domain multiple paths) as the egress point is the task of inter-domain/outbound TE. Once the egress point has been selected, say ASBR 10.10.10.1, intra-domain traffic engineering is then responsible for selecting the best intra-domain path between each pair of ASBRs in the network. In this simple example, intra-domain TE attempts to find an optimal internal path (or multiple paths if intra-domain multi-paths are allowed) from ASBR 10.10.10.3 to ASBR 10.10.10.1 as well as an optimal path C from 10.10.10.3 to ASBR 10.10.10.2.

Despite their clear difference in definition, intra- and inter-domain traffic engineering should not be considered independently of each other in practice, since the network configuration of one could potentially impact the other. Research has emerged recently on the interaction between the two types of TE, and some results are presented in [6][7]. We will provide more details on the interaction between intra- and inter-domain traffic engineering in Section 6.2.

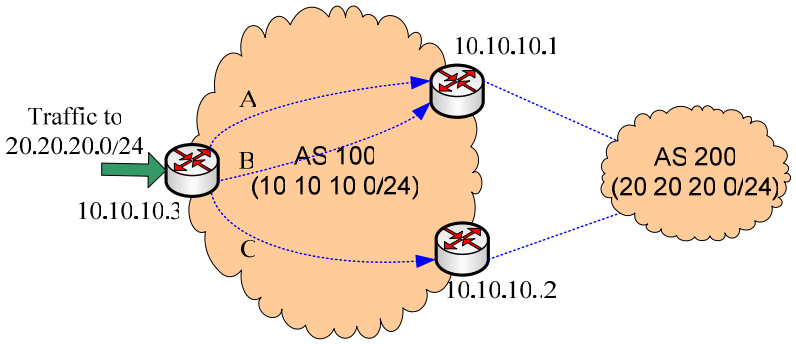

Figure 3. Intra- and Inter-domain traffic engineering

2.2 MPLS based TE vs. IP based TE

The concept of traffic engineering was first introduced in Multi-Protocol Label Switching (MPLS) based environments [4][8]. By intelligently setting up dedicated Label Switched Paths (LSPs) for delivering encapsulated IP packets, MPLS oriented traffic engineering can provide an efficient paradigm for traffic optimization. The most distinct advantage of MPLS oriented TE is its capability of explicit routing and arbitrary splitting of traffic, which is highly flexible for both routing and forwarding optimization purposes. However, since traffic trunks are delivered through dedicated LSPs, scalability and robustness become issues in MPLS oriented TE. First, the total number of LSPs (assuming full mesh or equivalent) within a domain is $O(N^2)$ where N is the number of ASBRs. This means that the overhead of setting up LSPs can be very high in large-size networks. In addition, path protection mechanisms (e.g., using backup paths) are necessary in MPLS oriented TE, as otherwise traffic cannot be automatically delivered through alternative paths in case of any link failure in active LSPs.

The first IP-based traffic engineering solution was proposed by Fortz et al [9]-[11]. The basic idea of their approach is to set the link weights of Interior Gateway Protocols (IGP) according

to the given network topology and traffic demand, so as to control intra-domain traffic and meet TE objectives. More recently, schemes that manipulate BGP routing attributes, known as BGP tweaking [12], have also been proposed for inter-domain traffic engineering. In comparison to the MPLS-based approach, these IP-based TE solutions lack flexibility in path selection, since explicit routing and uneven traffic splitting are not supported. However, the IP-based approach has better scalability and failure resilience than MPLS-based TE, because no overhead for dedicated LSPs is required, and also because traffic can be automatically delivered via alternative shortest paths in case of link failure, without explicitly provisioning backup paths. However, given this type of auto-rerouting in the IP based environment, recent research work [13] has suggested that a single link failure can introduce dramatic changes to traffic distribution even across multiple domains, as a significant proportion of traffic will switch to new shortest paths once the network topology has changed. This low TE robustness is in comparison to the MPLS TE schemes, where a single link failure does not impact other primary LSPs unless they are using the faulty link. Table 1 summarizes the key differences between MPLS-based TE and IP-based TE.

	MPLS oriented TE	**IP oriented TE**
Routing mechanism	Explicit routing with packet encapsulation	Plain IGP/BGP based routing
Routing optimization	Constraint based routing (CBR)	IGP link weight adjustment BGP route attribute adjustment
Multi-path forwarding	Arbitrary traffic splitting	Even traffic splitting only
Hardware requirement	MPLS capable routers required	Conventional IP routers
Route Selection flexibility	More flexible - arbitrary path	Less flexible – shortest path only
Scalability (overhead in maintaining network state)	Less scalable	More scalable, with scalability of underlying routing protocol
Failure impact on traffic delivery	High (normally need backup paths in case of failures)	Low
Failure impact on TE performance	Low	High

Table 1. MPLS/IP TE comparison

2.3 Offline TE vs. Online TE

The third part of our taxonomy is to classify traffic engineering into offline and online. As previously mentioned, the principal difference between offline and online traffic engineering is the availability of a traffic matrix (TM). The concept of traffic matrix was originally associated with intra-domain TE, where ingress/egress points of traffic are fixed. In this case, the overall traffic demand on the network can be represented by a matrix TM, e.g., with each element *t(i,j)* of the TM being the total bandwidth demand of all individual traffic flows (known as *traffic trunk*) from ingress node *i* to egress node *j*. When inter-domain traffic engineering is concerned, ingress/egress nodes for a traffic trunk might not be specified; instead the traffic is from some source (e.g an AS) to some destination (e.g. represented by a destination address or by a next-hop AS or a destination AS).

In some scenarios it is possible for an INP to forecast the traffic matrix before routing optimization is performed. Currently, there exist two principal inputs from which traffic matrix can be forecast: a Service Level Specification (SLS) and monitoring/measurement.

SLS is the detailed information on the agreement negotiated between customers and the INP. By aggregating the traffic predicted in the SLSs with individual customers, the INP can estimate the overall bandwidth demand between each pair of ASBRs. In addition, the INP can also apply monitoring/measurement mechanisms at the network boundary for aiding traffic matrix estimation. Having obtained the traffic matrix for the specific network topology, an INP can perform offline traffic engineering, i.e. map optimally the whole traffic matrix onto the physical network. Figure 4 presents a basic diagram for the offline TE process. One important issue in offline traffic engineering is the average duration between two consecutive TE cycles, and this period is known as Resource Provisioning Cycle (RPC) [14]. In common practice, the RPC for offline TE is weekly or monthly, depending on various factors such as the frequency of establishing, modifying and terminating SLSs with customers.

In some cases, an INP might not be able to predict the overall traffic matrix in advance, and this requires the INP to perform online traffic engineering that is blind to future traffic demands. In this scenario, the basic task of resource optimization is to optimally assign the newly incoming traffic one by one so that the possibility of accommodating further incoming traffic without congestion can be maximized. Towards this end, online TE approaches should make sure that the traffic load is as evenly distributed as possible within the network, so that random incoming traffic demand in the future can be easily satisfied. In some cases, it is also possible to reroute *existing* flows in the network so as to reserve bandwidth for the newly incoming traffic. However, this rerouting should not be performed large scale, as competing flows might interfere withd each other and cause traffic instability and service disruption.

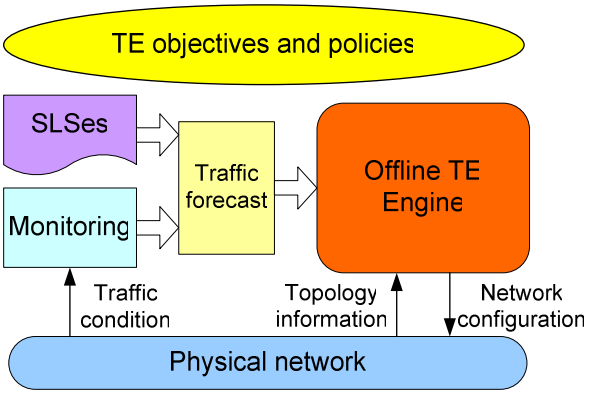

Figure 4 Offline traffic engineering

2.4 Unicast TE vs. Multicast TE

The Internet carries heterogeneous traffic, including both unicast/multicast traffic and various types of flows that use overlay routing techniques. In this article we survey not only unicast TE but also multicast TE, which is becoming important given recent progress in Internet multicast service development. Compared to unicast TE, multicast traffic engineering is more complicated, since multicast routing is associated with point-to-multipoint tree construction. In the literature, resource optimization in multicast TE is normally formulated as a Steiner tree related problem with the objective of minimizing bandwidth consumption. Although their TE problem formulations might be different, it should be noted that, since IP unicast and multicast traffic can be simultaneously injected into the same physical network, traffic engineering for both types of traffic should not be done independently, without an awareness of each other.

3. Intra-domain Traffic Engineering

In this section we focus on routing optimization algorithms for intra-domain traffic engineering. We first split intra-domain traffic engineering into MPLS-based and IP-based subsections, and within each of them we discuss both offline and online traffic engineering.

3.1 Intra-domain MPLS Oriented TE

3.1.1 MPLS Overview

Multi-Protocol Label Switching (MPLS) is an IETF (Internet Engineering Task Force) standardized forwarding scheme. In MPLS, traffic is sent along Label Switched Paths (LSPs). An LSP is the path between an ingress label switching router (LSR) and an egress LSR. At the boundary of an MPLS domain, LSRs classify IP packets into Forwarding Equivalence Classes (FECs) and append different labels for packet forwarding within the MPLS domain. The Label Distribution Protocol (LDP) [15] is used to distribute label bindings during the setting-up of an LSP.

MPLS is a powerful technology for Internet traffic engineering, as it allows traffic to be forwarded onto an arbitrary explicit route, which may not necessarily follow the shortest path computed by conventional IP routers. Typically, individual flows are aggregated by MPLS TE into traffic trunks identified by FECs, which are then carried on LSPs between ingress and egress routers. In this case, the conventional shortest path based routing infrastructure (e.g., OSPF) is overridden with MPLS explicit routing tunnels. In order to support traffic-engineered explicit routing of these flow aggregates, two types of end-to-end signaling protocols are commonly used for setting up and tearing down LSPs, namely RSVP-TE [16] and CR-LDP [17]. RSVP-TE is a soft-state signaling protocol that uses the RESV and PATH messages in the Resource reSerVation Protocol (RSVP) [18] for a two-stage process in setting up LSPs. CR-LDP is a hard-state signaling protocol that runs over TCP and uses LDP REQUEST and RESPONSE messages for setting up traffic-engineered paths. [17] specifies how to set up TE-aware LSPs using CR-LDP. In order to disseminate TE information (e.g., reservable bandwidth) so that all nodes in the network have a consistent view of the associated traffic-engineering parameters, TE-extensions to IGP, e.g., OSPF (OSPF-TE) [19] and ISIS (ISIS-TE) [20] have been proposed to disseminate TE-aware link state advertisement for establishing traffic engineered LSPs.

With the rapid deployment of Wavelength Division Multiplexing (WDM) based optical networks, Generalized MPLS (GMPLS) oriented traffic engineering is becoming popular. Interested readers can refer to [3] for traffic engineering in optical networks.

3.1.2 Components of MPLS-based TE

Before describing individual TE schemes that use MPLS, we highlight the fundamental components in an MPLS-based TE framework [8][21]. According to [4], there are three basic capabilities involved: (1) a set of attributes associated with traffic trunks, (2) a set of attributes for network resources and (3) constrained based routing (CBR) for path selection.

The task of the traffic trunk attributes is to describe the basic properties of traffic trunks. In general, these properties include ingress/egress LSRs, the FEC to which the traffic trunks are

mapped, and a set of characteristics associated with the traffic trunk, e.g., bandwidth demand. Resource attributes are used to specify the physical network that individual traffic trunks pass through. Constraint based routing is performed based on this set of attributes, to find a feasible path with sufficient bandwidth to support the traffic trunk. Within this attribute set, the Maximum Allocation Multiplier (MAM) attribute is a configurable parameter that determines the proportion of resources available to a specific traffic trunk. Resource Class Attributes are used to enable multiple policies with respect to both traffic and resource oriented performance optimization. By applying different resource class attributes, it is possible for INPs to partition network resources (e.g., bandwidth) for dedicated TE objectives within each class. Finally, the constraint based routing component offers a demand driven and resource reservation aware routing paradigm for traffic and resource optimization purposes. In [4], the difference between QoS routing and CBR is also specified: QoS routing is a subset of constraint based routing, which can cover both selfish routing from customer's point of view and traffic engineering from INP's perspective. In section 6 we will provide more discussions on the relationship between QoS routing and TE.

3.1.3 Offline Traffic Engineering

A generalized MPLS routing optimization can be formulated as a multi-commodity flow problem [22], and can thus be solved using linear programming for yielding an optimal solution for routing mechanisms that allow arbitrary traffic splitting. However, this approach is often regarded as impractical, especially in a large-sized network, since the number of LSPs required is potentially huge due to arbitrary traffic splitting. To obtain a more scalable TE solution, traffic splitting has to be limited in scope. An early MPLS TE approach used simple constraint-based routing (CBR) without coordination between individual traffic trunks [21]. A typical CBR algorithm is as follows. Before setting up an LSP for a specific traffic trunk, all the infeasible network links (e.g., those with insufficient available bandwidth) are removed from the network topology. Shortest path routing (SPR) is then performed on the residual network graph and the LSP is assigned to this shortest path. The algorithm repeats the above procedure until all the traffic trunks are assigned. This routing algorithm is known as Constrained Shortest Path First (CSPF). Other routing schemes have also been proposed to extend SPR, such as Widest Shortest Path (WSP) and Shortest Widest Path (SWP) [23][24], both of which try to increase the available bandwidth at bottlenecks along the path. By applying WSP/SWP, not only has the underlying traffic a higher probability of finding a feasible path, but also network bottlenecks are avoided by "reserving" bandwidth resources for future demands, benefiting other traffic from this more sophisticated routing strategy.

In the literature, many MPLS TE schemes have addressed the problem of minimizing the maximum utilization; this approach is often formulated as a linear- or integer-programming problem. In [25], traffic engineering is investigated using both single and multiple paths. The authors prove that TE with multi-paths (LSP bifurcation) and arbitrary splitting of traffic is able to achieve optimal solutions using linear programming, while integer programming can be applied to MPLS TE without LSP bifurcations. In [26], Mixed Integer Programming (MIP) can be applied for calculating LSP routing and traffic splitting ratios with hop count constraints and node/link preferences. The authors claim that by confining traffic splitting ratios to discrete values (e.g., 0.1, 0.2 etc.) that are more suitable for implementation of LSPs, near optimal solutions can be obtained for the task of minimizing the maximum link utilization. Similarly, MIP is also used in [27] for multi-objective MPLS traffic engineering, with minimum delay, optimum load balancing and minimum splitting of LSPs being key objectives.

With the development of Differentiated Services (DiffServ), DiffServ based MPLS traffic engineering has become a research area for supporting QoS differentiation. DiffServ-MPLS based TE is now supported by both Cisco and Juniper routers, with CSPF being the

fundamental routing algorithm. In addition, more sophisticated DiffServ aware/equivalent MPLS TE schemes have also been proposed in the literature [28]-[33]. From an LSP construction perspective, [28] proposed an integrated approach that combines the CSPF and WSP algorithms. In [29], one primary path and one secondary path are constructed between each ingress/egress node pair, with the primary being the minimum hop count path and the secondary being the *disjoint* second minimum hop count path. Thereafter, a traffic trunk is split across both paths using Available Bandwidth Rate (ABR)-like explicit rate feedback from the network. The authors of [32] proposed a general framework for intra-domain QoS provisioning through MPLS oriented TE in DiffServ networks. From a routing optimization perspective, the TE objectives are (1) to satisfy the QoS requirements of the traffic trunks, and (2) to minimize the overall network cost (load). The cost function is formulated as a convex function of the traffic load on per-QoS class basis, and the TE optimization task is formulated as a non-linear programming problem. In order to find the optimal solution, the authors apply a general gradient projection method for calculating LSPs. The QoS metrics considered in this work include end-to-end delay and loss, both of which are transformed into unified hop-count based constraints. In order to verify whether these QoS requirements are met during the optimization process, shortest path adaptations (e.g., k^{th} shortest paths) are applied on a hop-count basis. In [33], a Differentiated Traffic Engineering (DTE) solution was proposed. To solve the path selection problem in DTE, the overall routing optimization is decomposed into two sub-problems: the non-convex part of the optimization problem is solved by a simulated annealing technique, while the convex part is solved using the gradient projection method.

Apart from the pipe model, where LSPs are point-to-point, other papers have also proposed alternative models, such as the funnel model (multipoint to point, MP2P) [34]-[36] and the hose model (point to multipoint, P2MP) [37]. The advantage of these alternative models in LSP construction is to alleviate the scalability issues in LSP construction and maintenance. In order to reduce the total number of LSPs needed, the authors in [34] proposed a TE scheme using multiple MP2P LSPs. Specifically, the proposed approach consists of two distinct procedures, namely MP2P LSP construction and flow assignment. During the phase of LSP construction, a set of point-to-point paths is first selected between each ingress/egress pair with two constraints: (1) the total hop counts of each path should not exceed the threshold that is the hops of the minimum hop-count path plus a predefined number, and (2) at least one path must be node-disjoint with the rest of the path set. If such a path set cannot be found, then a path pair is selected comprising the minimum hop path and another disjoint path with a second minimum hop count. Thereafter, the MP2P LSP design applies binary integer programming on a per-egress router basis, and merges the pre-selected point-to-point paths. In the flow assignment phase, the task is to map the traffic trunks onto the constructed MP2P LSPs with the objective of minimizing the maximum load. In this work, the design of MP2P LSPs has three distinct advantages: LSP scalability, load balancing and resilience. In [35], MP2P LSPs are used for traffic engineering with deterministic end-to-end QoS guarantees. In addition, two admission control algorithms are introduced at the packet level, but routing optimization is not much addressed in this work. MP2P traffic engineering has also been studied in [36], where the scalability issue in MPLS label space is investigated. The basic idea is similar to [34], which attempts to merge point-to-point paths into MP2P LSPs. However, this work assumes that the P2P paths are pre-defined so that the task is only to assign each of them to individual MP2P LSPs. From this point of view, routing and resource optimization are not the major concern in this work.

A summary of published offline MPLS TE work is presented in Table 2.

Reference	Optimization Objectives/metrics	Optimization method	LSP type	Applicable environment
[25]	Minimize maximum utilization	Linear programming	P2P	Any
[26]	Minimize maximum utilization	Mixed Integer Programming (MIP)	P2P	Any
[28]	Minimize network cost with delay/bandwidth guarantees	Heuristic (CSPF + WSP)	P2P	DiffServ
[31]	Minimize network cost with QoS constraints	Non-linear programming (Gradient projection)	P2P	DiffServ
[32]	Minimize network cost across multiple classes	Simulated annealing + Gradient projection	P2P	DiffServ
[34]	Minimize the number of LSPs and hop-counts	Heuristic + binary integer programming	MP2P	Any
[35]	Provide deterministic end-to-end QoS	Not available	MP2P	Any
[36]	Minimize the overhead in LSP labels	Not available	MP2P	Any
[37]	Minimize LSP bandwidth allocation	Not available	P2MP	Any

Table 2. Offline MPLS TE solutions

3.1.4 Online Traffic Engineering

Online MPLS oriented traffic engineering can be classified into two categories: (1) dynamically adjusting the traffic splitting ratio among *pre-constructed* static LSPs [38][39]; and (2) computing dynamic LSPs on the fly for each new traffic trunk demand. MATE [38] is a typical example of the first category, and its basic operation is to adaptively forward incoming traffic onto multiple *pre-constructed* LSPs according to probing results from the network core. For this TE paradigm, routing optimization is not directly involved, as traffic and resource optimization is achieved through online forwarding adaptation. In the rest of this section we will restrict our focus on the second category of online MPLS TE.

The CSPF, WSP and SWP algorithms described earlier are the fundamental routing solutions that can be applied to online MPLS TE schemes. In DORA [40], the online TE solution contains two stages that maximize the ability of the network to accommodate future bandwidth-specified traffic demands. First, a parameter called Path Potential Value (PPV) is computed for each link on per ingress/egress node pair basis. The metric of PPV indicates the frequency with which each link has been used in the disjoint paths between ingress/egress node pairs. In the second stage, network links without sufficient residual bandwidth are removed from the network graph, and then a combined weight is calculated for each remaining link based on the PPV value and the available bandwidth, with a tuning parameter known as BWP (bandwidth proportion) for handling the tradeoff between the two metrics. Finally, a conventional Dijkstra's shortest path algorithm is applied based on the set of defined link weights.

One important issue that is often addressed in online MPLS TE schemes is the LSP interference problem [41]-[44]. The authors of [41][42] noticed that, by directly setting up

LSPs (e.g., using CSPF) without considering the location of ingress/egress nodes for incoming traffic trunks, potential congestion is liable to take place at some critical links which multiple LSPs use. Competition by LSPs on the critical links that do not have sufficient available bandwidth for supporting all the LSP demands is known as LSP interference. Figure 5 provides a simple example of this. First, we assume an incoming traffic trunk from ingress node D to egress node G. If this is assigned the shortest-path based LSP (D→E→F→G), then future traffic trunks from H to I will be blocked if the residual link (E, F) cannot support both demands. In effect, we can find from the network topology that link (E, F) is *critical* to the traffic trunks from H to I in that any LSPs from H to I need to use that link. In this case, a more intelligent strategy is to route the traffic trunk from D to G via an alternative longer path (D→A→B→C→G) and reserve the bandwidth on the critical link (E, F) for the future demand from the traffic trunk from H to I. From this example we can see that critical links are associated with the location of individual ingress/egress pairs. Hence, if the location of the ingress/egress nodes for traffic trunks is taken into consideration, then the probability of LSP interference can be decreased, if the LSP construction bypasses the critical links. Towards this end, the authors proposed the Minimum Interference Routing Algorithm (MIRA) so as to defer high loading on critical links. First, critical links associated with individual ingress/egress-pairs are identified through calculating the maxflow value. Thereafter, an ingress/egress-pair specific weight is created for each link, being an increasing function of its criticality. Finally, conventional shortest path algorithms are used according to the resulting link weights on top of the network graph containing only feasible links that can support the bandwidth demand of the incoming traffic trunk. The authors also implemented a software package called Routing and Traffic Engineering Server (RATES) [45], which is based on MIRA. Further, in [43], the authors enhanced the MIRA algorithm by taking into account the overall blocking probability of LSP demands. Their scheme is based on the observation that MIRA only focuses on the interference between one single pair of ingress/egress routers, but it is not able to deal with the critical links associated with multiple ingress/egress pairs.

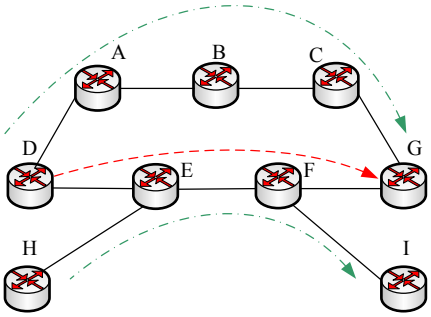

Figure 5. LSP interference

Online MPLS traffic engineering has also been studied in DiffServ environments for QoS support, a typical example being TEAM [44]. The Traffic Engineering Tool (TET) in the TEAM framework is responsible for LSP preemption and construction. First, for each incoming demand, three types of costs are considered in the cost function, namely bandwidth cost, switching cost and signaling cost. The objective of LSP manipulation is to minimize the overall cost throughout the process, which can be achieved by a Markov process based decision. There exist two distinct LSP operations in TEAM: LSP pre-emption and LSP routing. LSP preemption allows existing LSPs to be preempted by newly constructed LSPs with higher priority. To do this, each LSP is assigned a priority attribute, which is taken into account when there exists competition for resources (i.e., interference). Thus, even if an LSP has already been assigned a path, it will be rerouted if it has a lower priority attribute than a new LSP that is competing for the shared network resources. In order to avoid frequent LSP switching and thus traffic instability, the proposed pre-emption policy include the following

three guidelines: (1) pre-empt the LSP with the lowest priority attribute, (2) pre-empt the fewest number of LSPs and (3) pre-empt the least amount of bandwidth while satisfying the traffic demand requirement. For LSP routing, the Stochastic Performance Comparison Routing Algorithm (SPeCRA) [46] is adopted in TEAM. SPeCRA behaves like a homogeneous Markov chain where the optimal routing scheme is a state of the chain that is visited at the steady state. Specifically, it attempts to select adaptively the best routing algorithm from a set of candidate schemes, each of which might be suitable for a specific type of traffic trunk. The same authors also proposed a new DiffServ-based LSP pre-emption policy known as V-PREPT that attempts to avoid LSP rerouting [47]. Similar to the TEAM scheme, the optimization for LSP pre-emption considers multiple criteria, including LSP priority, the number of LSPs and the pre-empted bandwidth. With V-PREPT, the tradeoff between the three criteria can be adaptively tuned according to the policy adopted by the INP. Apart from the simple LSP preemption algorithm, an adaptive version of V-PREPT was also proposed for reducing the overhead (essentially in signaling) introduced by frequent events of LSP teardown and rerouting. The basic idea of the adaptation is to allow some LSPs with lower priority attributes to have their rate allocation reduced so as to accommodate more requests in the future. In this case, RSVP-TE signaling is responsible for indicating the updated allocation of rate on the static LSP, while there is no extra signaling overhead in tearing down and setting up LSPs. In DiffServ based networks, this adaptive V-PREPT scheme is useful in LSP operations for the Assured Forwarding (AF) per hop behavior (PHB). Give the common practice that the Expedited Forwarding (EF) behavior is normally used to support hard QoS guarantees, bandwidth allocation in AF PHBs can be more flexible and dynamic, and the proposed adaptive V-PREPT algorithm can be efficiently adopted for this class of PHBs.

Survivable online traffic engineering in MPLS networks has also been considered in [48]. Similar to MIRA, this scheme constructs LSPs dynamically by applying the shortest path algorithm to the dedicated link weight metric that reflects the specific TE requirement. This type of dynamic link metric is based on a Lost Flow in Link (LFL) function that is used to assign working routes with local restoration. In LFL, the metric of a particular link reflects the change in the objective function if an incremental demand has been (re)routed through or even near that particular link.

A summary of the existing online MPLS TE approaches is shown in Table 3.

Reference	Optimization Objectives/metrics	Major LSP computing method	Applicable environment
[40]	Maximize future traffic demands accommodation with bandwidth guarantees	Heuristic (CSPF based)	Any
[41] [42] [43]	Minimize LSP interference so as to accommodate maximum future demands	Heuristic (CSPF based)	Any
[45]	Minimize bandwidth, switching and signaling costs	The SPeCRA algorithm	DiffServ
[47]	Optimize LSP priority, number of LSPs and preempted bandwidth	V-PREPT for LSP preemption	DiffServ
[48]	Minimize loss of traffic flow	Heuristic (k^{th} shortest path based)	Any

Table 3. Online MPLS TE solutions

3.2 Intra-domain IP Oriented Traffic Engineering

3.2.1 Theoretical Background

The advent of plain IP oriented TE solutions has recently challenged MPLS-based approaches in that Internet traffic can also be effectively tuned through native hop-by-hop based routing, without the associated complexity and cost of MPLS. In [49], the authors proved that any arbitrary set of loop-free routes can be resolved into shortest paths with respect to a set of positive link weights that can be calculated by solving the dual of a linear programming formulation. This implies theoretically that, if a network is optimally engineered through a set of loop free explicit LSPs, by setting appropriate OSPF/ISIS link weights, this set of LSPs can be transformed into shortest paths according to this set of link weights. As a result, plain IP routers can directly compute this set of paths by using Dijkstra's algorithm, and hence the associated LSPs are not necessary anymore. Take the small network in Figure 6(a) as a simple example (with symmetric weight setting in both directions of each link): The explicit path set {a→c→b, b→c→d} are shortest paths if we assign the weight value of 3 to links (a, b) and (b, d), and set the weight of all the other links to be 1. Nevertheless, there are two major issues that restrict the practical deployment of link weight optimization based traffic engineering. First, not any arbitrary set of paths can be represented into shortest paths according to a set of link weights. For example, if we add another explicit path d→b→c to the aforementioned path set, as it is shown in Figure 6(b), these three paths cannot be represented simultaneously into shortest paths with any set of link weights, as the two paths b→c→d and d→b→c form a path cycle. Second, the distinct advantage of MPLS based TE is not only explicit routing, but also arbitrarily unequal splitting of traffic. In this case, even if a set of LSPs can be represented into shortest paths, it is still not possible to unequally split the traffic given the underlying OSPF/IS-IS routers. Evolving from [49], [50] presented further analysis on the relevant issues in *shortest path representability*. One important contribution from this work is how to prevent unintended paths from becoming shortest paths when setting specific link weights. The authors argue that the network could suffer from traffic sub-optimality if some bad paths are included in the shortest path set that will be configured to deliver customers' traffic.

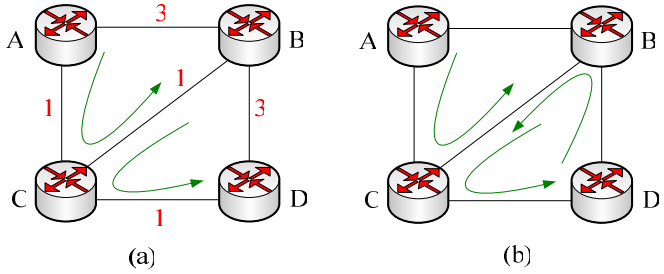

Figure 6. Shortest path representation

3.2.2 ECMP based Link Weight Optimization

In the Equal Cost Multi Paths (ECMP) mechanism, if there exist multiple shortest paths with equal IGP link weights towards the same destination, traffic is *evenly* split onto the next hop routers on these paths. Normally, the forwarding behavior in ECMP is on per-flow basis rather than a per-packet basis so as to avoid out-of-order packet arrival. This multipath approach was first adopted and analyzed in the Netscope TE tool [51].

Fortz and Thorup [9]-[11] claimed that by optimizing OSPF/IS-IS link weights for the purpose of load balancing, the network service capability can be improved by 50% to 110% in comparison to the conventional configuration of link weight setting using inverse proportional bandwidth capacity. The key idea of the proposed algorithm is to adjust the weight of a certain number of links that depart from one particular node, so that new paths with equal cost are created from this node towards the destination. As a result, the traffic originally traveling through one single path can be evenly split into multiple paths with equal OSPF/IS-IS weights based on ECMP. In general, the authors proved that the optimal configuration of such link weights is NP-hard. Figure 7 provides a simple illustration of the basic idea of the algorithm. Consider destination node *t* and assume that part of traffic demand going to *t* travels through an intermediate node x. The Fortz and Thorup's strategy is to spit the flow to *t* going through *x* evenly along all the links (x, x_i) from x, if these links (x, x_i) belong to the shortest path from *x* to *t*. This type of "local adjustment" needs special attention, since shifting traffic might incur additional congestion to other links. In order to avoid this oscillation phenomenon, the authors apply sophisticated Tabu search for achieving the best load balancing performance.

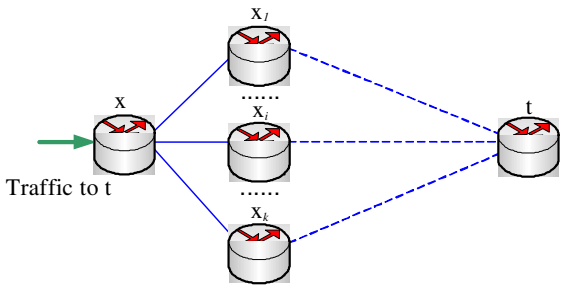

Figure 7 Fortz and Thorup's link weight optimization algorithm

[52] also proposed a Genetic Algorithm (GA) based approach for the same IP traffic engineering optimization problem, and the authors claimed that, by properly tuning the GA parameters, the resulting performance is very close to that of [9]-[11]. Retvari et al. additionally raised some practical issues in OSPF traffic engineering, e.g., explicit knowledge of link capacity and reasonable range of OSPF link weight values [53]. Towards this end, the authors formulated the traffic engineering as the Prime Minimum Cost Maximum Throughput problem, and the resulting link weight configuration provides a plausible basis to build a practical IP oriented TE architecture.

Optimal routing often requires arbitrary traffic splitting. Instead of optimizing OSPF/ISIS link weights, another TE approach for near-optimal network performance is to emulate uneven traffic splitting over ECMP paths at the edge or core routers. In [54], the authors proposed a scheme based on the manipulation of a subset of next hops for some routing prefixes; the scheme is capable of achieving near optimal traffic distribution without any change of existing routing protocols and forwarding mechanisms. The basic idea behind is as follows. First, optimal link weights are calculated based on [49] through linear programming. Second, in order to deal with the requirement of arbitrary traffic splitting, the authors proposed activating only a subset of ECMP next hops for packet forwarding to the selected destination prefix so as to emulate unequal splitting of traffic in the MPLS based solutions. Three different heuristic algorithms were studied for optimally configuring the next hop of unicast destination prefixes. This approach exhibits a typical strategy of making graceful trade-off between the performance and the overhead associated with the additional configuration needed.

3.2.3 Edge based Link Weight Setting

Wang et al. proposed in [55] a new OSPF traffic engineering approach without the necessity of ECMP splitting. Their approach is to divide the physical network into several logical routing planes, each being associated with a dedicated link weight configuration. There are two distinct procedures involved. First of all, the overall external traffic demands from all customers are partitioned properly into *k* traffic matrices only at the edge of the network, and each of the traffic matrices is identified by the Type of Service (ToS) or Differentiated Services Code Point (DSCP) in the IP header. Second, individual traffic matrices are independently routed over the *k* planes, each of which has its dedicated link weight configuration. The basic strategy of this approach is to emulate MPLS unequal splitting of flows by partitioning the overall traffic demand at the edge of the network so that traffic within different partitions is delivered through dedicated routing planes. To achieve the best overall traffic distribution, one of the most challenging tasks is to efficiently assign flows to the traffic matrices for different planes. Through simulations, the authors prove that a fairly small number of overlays (*k* equal to 2 or 4) can achieve near-optimal traffic engineering performance.

Table 4 presents a brief comparison of the IP oriented TE approaches.

Reference	Feasibility	Traffic splitting	Protocol requirement	Configuration complexity	Performance
[49] [50]	Theoretical analysis only	Arbitrary splitting	-	-	Theoretically optimal
[9]-[11] [52]	Practical	ECMP	Plain IGP	Conventional IGP link weight setting	50-110% improvement
[54]	Practical	Selective ECMP	Plain IGP	Manual configuration of next-hops for some prefixes	Near-optimal
[55]	Practical	Traffic splitting at the network edge	ToS-aware routing with multi-RIB IGP	Need configuration of multiple sets of link weights	Near-optimal

Table 4. IP Oriented TE solutions

3.2.4 Online IP-based Traffic Engineering

Unlike offline TE, which has been extensively studied, there also exist few proposals for online or adaptive IP-based TE. Two online TE approaches are to change link weights on the fly and to make link weights sensitive to some loading or QoS parameters (e.g. to make the link weight a function of link utilization or delay). However, these approaches require the flooding of new link weights throughout the network, which can cause route instability and looping problems during the convergence process [56].

Another online TE approach is to dynamically adjust the traffic splitting ratio according to the network load. OSPF-OMP (Optimized MultiPath) [57] was proposed to adjust the traffic

splitting ratio gradually over multiple relaxed shortest paths (non equal cost) by modifying the hash function, based on the loading information inside the network distributed by the OSPF Opaque LSA (Link State Advertisement) option. As with OSPF-OMP, Adaptive MultiPath (AMP) [58] considers multiple non-equal cost paths and balances load by optimizing the traffic splitting ratios at each router. However, AMP only keeps network available information to a local scope rather than employing a global perspective of the network in each node.

4. Inter-domain Traffic Engineering

In this section we introduce inter-domain traffic engineering, an emerging topical research area that has evolved from its intra-domain counterpart.

The Internet is a large decentralized inter-network composed of more than eighteen thousand ASes or domains. From a business perspective, the relationship between any two domains can be classified into one of the following two types:

- *Transit service (customer-provider relationship)*. This type of relationship exists commonly between low- and high-tier INP networks. Low-tier INPs (typically stub domains) purchase transit services from higher-tier INPs for Internet connectivity.
- *Peering*. This type of relationship exists commonly between neighboring INPs that are roughly equal in size and at the same tier. The INPs agree to simply exchange traffic without making any payment to each other.

We can also classify all the domains in the Internet into two categories, namely *transit domains* and *stub domains*. Transit domains offer transit services, i.e. inter-domain traffic delivery across the Internet. Stub domains, on the other hand, are the leaf domains of the AS-level hierarchy. They only send or receive traffic, and do not provide transit services to any other AS. In general, the two types of domain have different inter-domain traffic engineering objectives. The incentive for transit domains to perform inter-domain traffic engineering is normally to optimize network resources so as to maximize their incoming revenue. On the other hand, stub domains compose more than 80% of ASes in the Internet and most of them are *multi-homed*. Hence, their principal inter-domain issue is how to minimize the monetary expense of subscribing to Internet transit services from their INPs.

Another dimension for categorizing inter-domain traffic engineering is *inbound* and *outbound* traffic engineering, which focus respectively on how to control inter-domain traffic entering or leaving a domain. A domain may only require either inbound or outbound traffic engineering, or both according to its business objectives. For example, a domain that contains popular content providers generates a large amount of traffic that needs to be sent out of the network efficiently, and thus outbound traffic engineering is needed. On the other hand, domains that have a large number of multimedia application receivers (e.g., Internet TV/MP3 subscribers) are typically traffic consumers. They therefore need to perform inbound TE in order to control traffic injected into their networks. Finally, since transit domains normally exchange Internet traffic between each other, both inbound and outbound TE may be required.

In the rest of this inter-domain TE section, we first give a brief introduction to the *de facto* inter-domain routing protocol, BGP-4 [59], which can be used to perform inter-domain traffic engineering by appropriately adjusting route attributes. Then, some general guidelines for inter-domain traffic engineering are presented. We then describe relevant TE work, classifying it into inbound and outbound traffic engineering. Finally we discuss advanced inter-domain TE paradigms, e.g., cooperative TE between adjacent domains.

4.1 BGP Overview

The Border Gateway Protocol (BGP) is brieflydescribed here. ASes interconnect with each other via dedicated inter-domain links or Internet Exchange Points (IXPs). Border routers from different ASes exchange routing reachability advertisements through external BGP (eBGP) sessions, and these advertisements are also propagated to all the rest of BGP speakers within the AS through internal BGP (iBGP) sessions. BGP allows *attributes* to be associated with each advertisement. BGP itself is a distance vector based routing protocol with a set of dedicated import/export policies that allow INPs to control inter-domain routes. In BGP routing, all the policies are prioritized lexically to form a sequential inter-domain path selection process. The BGP routing decision steps are described in Figure 8.

Many recent publications have described the inefficiencies of BGP, and some alternative solutions such as HLP [60] have also been proposed. Nevertheless, BGP is likely to remain as the de facto inter-domain routing protocol in the near future.

(1) Accept the advertisement with the highest *local-preference*;

(2) Break ties by accepting the advertisement with shortest *AS paths*;

(3) Break ties by preferring the route with the lowest *origin type*;

(4) Break ties by accepting the advertisement with the lowest *Multi-Exit-Discriminator* (*MED*) coming from the same neighboring AS;

(5) Break ties by preferring an *external* BGP advertisement over an *internal* one;

(6) Break ties by accepting the advertisement with the lowest intra-domain *IGP weight* to the egress router;

(7) Break tie by accepting the advertisement with the lowest *next-hop address*.

Figure 8 BGP path selection process

As described above, inter-domain traffic engineering can be classified into inbound/outbound TE, and an INP can configure BGP attributes so as to help achieve its TE objectives (see Tables 5 and 7). From Figure 8, it is obvious that only one single path should be selected for a particular destination prefix, because the final step of tie breaking is based on the unique IP address of the next hop of BGP peer. Some vendors have also implemented the BGP multi-path functionality. In Cisco's BGP implementation, if the INP chooses to enable BGP multi-paths, the tie-breaking criteria in steps 6-7 in the above process are overridden [61], which means that multiple (up to 6) inter-domain routes can be installed simultaneously into the BGP routing table for the same destination prefix. Similar to the intra-domain scenario, this BGP multi-path functionality provides flexible mechanisms for the INP to perform load balancing for transit traffic traveling through the network.

4.2 Inter-domain TE Guidelines

Inter-domain traffic engineering is performed by taking into account the routing information advertised by adjacent domains. We note that the change of TE configuration in one domain might affect the routing decisions of other ASes nearby, and this can propagate in a cascaded fashion. This often introduces route instability problems across the whole Internet, where a single change of inter-domain path may take up to several minutes to converge [62]. As a

result, domains may be unable to predict whether their inter-domain TE solutions can produce the target performance. Thus, inter-domain TE should take into consideration how to preserve its predictability as well as stability so as to ensure stable traffic distribution and fast routing convergence [63]. For this purpose, recent research has proposed several guidelines for inter-domain traffic engineering. We summarize the guidelines proposed in [62],[64] as follows:

- *Achieving predictable traffic flow changes*. The objective is to minimize the frequency with which upstream domains need to switch their outgoing traffic to different domains by changing the local BGP configuration. This adversely affects the traffic volume entering their networks.
- *Limiting the influence of neighboring domains*. The objective is to minimize the impact on routing decisions of neighboring domains. These routing decisions may contain inconsistent route advertisements from adjacent domains, which reduce the operator's control capability over traffic flows.
- *Reducing the overhead of routing changes*. If the traffic has to be separately engineered for all address prefixes in the Internet, the configuration overhead is too high to be realistic. To reduce this overhead, the number of destination prefixes to be considered should be limited through efficient address aggregation. In effect, it is suggested that INPs need to only engineer the traffic towards a small number of popular destination prefixes that accounts for a large portion of Internet traffic [64]. This TE strategy allows INPs to control efficiently a large portion of traffic in the Internet by considering only a small number of prefixes.
- *Customer routes preferred*. [62] has shown that Internet stability can be achieved by imposing a set of policies on individual domains. Thus, global coordination among all domains across the Internet is not necessary. The guidelines proposed in [62] ensure stable TE with fast convergence by favoring routing via customer domains over peer and provider domains. If customer domains are not directly available, then routing via peer domains is preferred over provider domains.

4.3 Outbound Traffic Engineering

4.3.1 Outbound TE Mechanisms

A number of mechanisms are currently known for outbound traffic engineering, as shown in Table 5.

Mechanism	Description	Implementation Techniques	Applicable Environment
BGP Local Preference (*local_pref*)	To select directly the egress router by setting the highest BGP local-preference value	BGP	Stub / Transit domains
Hot Potato Routing	To select the egress router with the lowest IGP weight	BGP/IGP	Usually Transit domains
Explicit routing (MPLS)	To select egress router by establishing explicit paths across domains	RSVP-TE BGP/IGP-TE PCE	Stub / Transit domains

Table 5. Mechanisms for outbound inter-domain TE

- *Setting Local-preference (local_pref).* The *local-preference* attribute has the highest priority in the BGP route selection process. The value assigned to this attribute indicates the preference on one border router to other candidates as the best egress point. Take Figure 3 as an example. If the local preference value for the prefix 20.20.20.0/24 on the border router 10.10.10.1 is higher than that on 10.10.10.2, then the traffic destined for AS 200 will use 10.10.10.1 as the egress point in AS 100.

- *Hot Potato Routing*. If multiple routes exist with equal value of BGP route attributes up to step 5 of the BGP route selection process shown in Figure 8, the route with the lowest IGP weight from the ingress to the egress point is selected. This scenario is known as hot potato or early-exit routing, which is often adopted by large INPs. The objective of hot potato routing is to send the traffic to downstream domains across the core network as quickly as possible. By manipulating IGP link weights an INP is able to influence egress router selections within the local domain. In Figure 3, we now assume that all the route attributes are "equally good" (Figure 8 steps 1 to 5) for both 10.10.10.1 and 10.10.10.2. If the IGP weight of shortest path A (between 10.10.10.3 and 10.10.10.1) is lower than that of shortest path C (between 10.10.10.3 and 10.10.10.2), then 10.10.10.1 is selected as the egress point according to hot potato routing.

- *Explicit routing (inter-domain MPLS).* Inter-domain MPLS enables a domain to enforce traffic to be delivered on the explicit paths to the destination across downstream domains [65]. Thus, domains may establish explicit paths through their desired egress points to the downstream domains and destinations. Currently, mechanisms supporting inter-domain MPLS have been proposed and implemented, e.g., Path Computation Element (PCE), and commercial products exist, for example from Cisco Systems.

4.3.2 Offline Outbound Traffic Engineering

We initially consider offline outbound traffic engineering in stub domains. The authors in [66] propose offline optimization algorithms to distribute the traffic of a multi-homed stub domain among multiple downstream INPs. The TE objective is to optimize both monetary expense and network performance (measured by average latency). The authors found that the optimization of expenses and performance are often in conflict. In order to cope with this, they consider an approach that tackles the expense and performance optimization separately and sequentially. First of all, the optimization of monetary expense is performed. This is based on the business operation viewpoint that minimizing the overall expense has higher priority than optimizing the network resource utilization in stub domains. Based on a percentile-based charging model, the objective of the optimization is to determine the amount of traffic to be sent to each of the downstream INPs so that the total charge is minimized. The performance optimization is then applied to assign the traffic to the downstream INPs. As a result, the total latency is minimized within the constraint of the computed expense. Instead of tackling the expense and performance optimization in a lexicological importance order, the authors in [67][68] propose a multi-objective evolutionary algorithm to solve a similar optimization problem. The aim is to find a compromising solution that is good with respect to all the optimization objectives. As with [66], the metric to be minimized is the charge incurred by the downstream INP, whereas the performance to be optimized is the load balancing across the inter-domain links. In addition to these two objectives, the authors also consider how to minimize the iBGP communication overhead in order to enforce the TE decisions. The authors in [69] introduced an INP subscription problem of subscribing to a set of downstream INPs so as to minimize the cost in payment. The INP subscription problem is different from the abovementioned expense optimization in that the latter assumes that the INP subscription decision has already been made, thus traffic can only be assigned to the

subscribed downstream INPs. However, in order to further minimize the monetary expense, a domain may have the freedom to select the optimal set of downstream INPs from all the available candidates and then assign traffic to this set of INPs. The INP subscription problem is based on a percentile-based charging model and is solved through dynamic programming. The authors in [70] addressed a similar INP subscription problem on top of a total-volume based charging model. Their work goes one step further in that that the chosen downstream INPs also need to provide end-to-end bandwidth guarantees towards the destination domains. The problem is solved by a Genetic Algorithm based approach.

We now describe a number of schemes that focus on transit domain traffic engineering issues. The BGP traffic engineering approach proposed by Bressoud et al. [71] was the first piece of work dealing specifically with outbound inter-domain TE for transit domains. The objective of the TE problem is to determine an optimal set of egress points for the advertisement of destination prefixes so as to minimize the traffic cost (i.e., bandwidth consumption) while satisfying the bandwidth capacity constraints of the inter-domain links. The outbound inter-domain TE problem is further subdivided into two parts: Single Egress Selection (SES) and Multiple Egress Selection (MES). SES ensures that one and only one egress point is selected for each destination prefix, whereas MES allows multiple egress points instead. Two heuristic algorithms, combining the approximation algorithm proposed for the Generalized Assignment Problem (GAP) with a simple greedy heuristic, were proposed to solve the SES and MES problems. Furthermore, the authors in [72] proposed two heuristic algorithms for the SES and MES that are more computationally efficient and able to obtain better TE performance. Finally, the authors in [75] proposed an open source tool, called Tweak-it, for outbound inter-domain TE in large transit domains. The authors in [73][74] extended outbound inter-domain TE so as to support end-to-end bandwidth guarantees across transit domains. Their work is based on the MESCAL cascaded model that allows negotiations between adjacent domains and achieve bandwidth guarantee by establishing INP-level SLAs [76]. As Figure 9 shows, each domain offers its upstream neighbor (through provider SLAs) a guaranteed bandwidth (o-BW) towards each destination aggregate prefix (Dest). Each SLA is associated with the amount of available bandwidth that is guaranteed from the offering downstream domains to the destination domains. In order to provide end-to-end bandwidth guarantees for the traffic, the outbound inter-domain TE problem has been extended for not only finding an optimal egress point that maintains the capacity constraints of inter-domain links and SLAs, but also the paths *within* the network to satisfy the traffic demand requirement. In [73], the TE objective is to minimize the total bandwidth consumption in the network, and the authors extended the problem to optimize multiple objectives in [74] - not only minimizing the total bandwidth consumption but also balancing the load over intra- and inter-domain links. Both problems can be formulated as an extended problem of egress router selection. The authors in [77] propose an inter-domain traffic engineering system for provisioning end-to-end delay guarantees in addition to meeting bandwidth requirements.

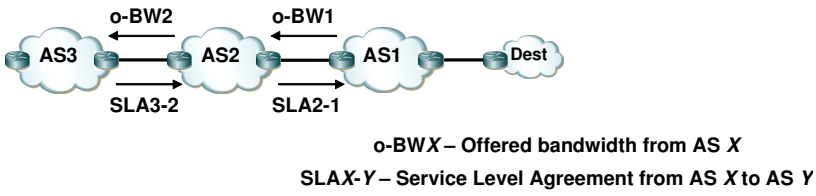

Figure 9 Cascaded model for end-to-end bandwidth guarantee

4.3.3 Online Outbound Traffic Engineering

In the literature, online outbound TE schemes have only focused on stub domains. They can be classified into the following two types:

- *Proactive*: the TE solutions rely on traffic predictors to forecast traffic on a short time interval (e.g. minutes), and then run a lightweight TE algorithm in a *quasi-offline* manner to produce solutions in short time scale.
- *Reactive*: the TE solutions are adaptive and dynamic to incoming traffic demand without traffic prediction beforehand.

In [66], the authors propose proactive online algorithms for multi-homed domains to select appropriate INPs for outbound traffic. The objective is first to minimize the total expense and then to minimize the end-to-end latency. The approach for the short-term traffic forecast is based on the exponentially weighted moving average (EWMA) method. In this scenario, traffic prediction is performed through detecting traffic changes based on a sequence of independent preceding observations. The proposed online TE algorithm is a greedy heuristic based on traffic sorting, which has also been used for solving the bin-packing problem [78]. Another proactive online TE approach was addressed in [79]. The authors designed a systematic BGP-based outbound TE technique for stub domains over the timescale of minutes. Apart from the TE objectives considered in [66], [79] also investigates how to minimize the overhead of the associated iBGP message advertisements. A quasi offline multi-objective evaluation algorithm was proposed to solve the online outbound TE problem.

For reactive TE paradigms, the first work on quantifying the benefits of dynamic route selection with multi-homing was proposed in [80]. The multi-homed domain under consideration may subscribe to multiple downstream INPs, and it also measures the end-to-end path performance (turn-around delay) through each downstream INP towards the destination. Based on the performance obtained from measurement, the domain dynamically switches traffic to the INP that has the best instant performance. Compared to random selection of INPs, the measurement-based multi-homing approach can achieve a 40% performance improvement in terms of the average turnaround delay. Based on this approach, the authors in [81] proposed a Round Trip Time (RTT) measurement approach for outbound route selection. The proposed approach is scalable and does not require RTT measurements via each INP to individual large number of destinations.

To summarize the outbound traffic engineering schemes in this section, we list and compare in Table 6 the major characteristics of the solutions that have been presented in this sub-section.

Reference	Optimization Objectives/metrics	TE Semantic	Implementation Techniques	Applicable Environment
[66]	Minimize overall expenses and end-to-end latency	Offline /Online	Not specified	Stub
[67][68]	Minimize overall expenses, improve inter-domain load balancing and minimize BGP communication overhead	Offline	Local_pref	Stub
[69]	Minimize overall expenses	Offline	Not specified	Stub
[70]	Minimize overall expenses and provide end-to-end bandwidth guarantee	Offline	Not specified	Stub
[71]	Minimize network cost (e.g., bandwidth consumption)	Offline	Local_pref, AS-Path	Transit
[73][74]	Minimize network cost and provide end-to-end bandwidth guarantee	Offline	Not specified	Transit

[79]	Minimize overall expenses, improve inter-domain load balancing and minimize iBGP communication overhead	Online	Local_pref	Stub
[80]	Turn-around delay	Online	Not specified	Stub
[81]	Round Trip Time (RTT)	Online	Local_pref	Stub

Table 6. Outbound traffic engineering approaches

4.4 Inbound Traffic Engineering

4.4.1 Inbound TE Mechanisms

In this section we first provide an overview of available mechanisms for inbound traffic engineering. As with outbound TE, although there exist various candidate implementation mechanisms, inbound TE routing optimization algorithms have only used a few of them, e.g., AS path prepending. Nevertheless, we list all of the potential mechanisms in Table 7 based on which inbound TE can be performed.

Mechanism	Description	Implementation Techniques	Applicability Environment
Selective advertisement	Advertise a route only at the set of ingress points that is expected to receive traffic	BGP	Stub / Transit
More specific advertisement	Advertise routes with more specific prefixes, to suppress the coarse-grained ones	BGP	Stub / Transit
AS-path prepending	Inflate the length of the AS path attribute to reduce the attractiveness of the route	BGP	Stub / Transit
Lowest MED value	Advertise preferred routes with the lowest value of MED	BGP	Stub / Transit
Community attribute	Suggest to adjacent domains how to manipulate the advertised routes	BGP	Stub / Transit
Network Address Translation	Modify the packet headers by assigning the desired ingress point as the source of packets	NAT	Usually stub
BGP Overlay	Direct communication between any two domains bypassing BGP	User specified	Stub / Transit

Table 7. Mechanisms for inbound inter-domain TE

- *Selective advertisement*. In this approach, routes towards a destination prefix are only advertised through a set of chosen ingress links. We take Figure 10 as an example. If AS300 would like to receive traffic from AS400 via ASBR 30.30.0.1 heading towards AS301, it chooses not to advertise the route to AS301 through ASBR 30.30.0.2. However, the shortcoming of this approach is that if the chosen ingress point fails, no alternative routes can be used as backup.
- *More specific advertisement*. In this approach, if multiple routes exist towards the same destination, the one with the longest-matching prefix will be selected. In Figure 10, we

assume AS300 advertises to AS400 the reachability of destination prefix 30.30.0.0/16 on 30.30.0.1, and its sub-prefix 30.30.30.0/24 on 30.30.0.2. As a result, the traffic towards any destination in "nested" AS301 will not use 30.30.0.1, as the other ingress router has a route with more specific prefix. Compared to selective advertisement, this type of ingress point selection is more robust in case of link failure. If the inter-domain link attached to 30.30.0.2 breaks, the traffic towards AS301 can still be routed via 30.30.0.1 using the route with more coarse-grained prefix.

- *AS-path prepending*. A route advertisement is made less attractive to upstream domains by adding several instances of AS-number to the AS-path attribute so as to inflate the AS-path length of that route. In Figure 10, if AS300 would like to receive traffic from AS400 towards AS301 via ingress point 30.30.0.1, then it may prepend its own AS number in the advertisement on 30.30.0.2, such that the overall AS path via this ASBR is made "longer" than via 30.30.0.1. It should be noted that, this is only possible if AS400 does not apply the local-pref metric to select the preferred route. Related work on and performance evaluation of AS-path prepending can be found in [82]-[84].
- *Setting MED value*. This applies only if two ASes have two or more direct connections between them and both ASes agree to implement MED. In these circumstances a domain may select its preferred ingress router by assigning a lower MED value. Consider the example of Figure 10, if AS300 would like to receive traffic from AS400 via 30.30.0.1, it may advertise BGP route with lower MED value through this router than the one on 30.30.0.2. The prerequisite for using the MED metric for ingress point selection is that all the route attributes with higher BGP route selection priority for the two routes should be set equal (e.g., the local_pref metric set internally by AS 400 and the AS path length via the two border routers).
- *Community attribute*. In this approach, a route can be advertised associated with the community attribute that instructs upstream domains how to manipulate this route with certain actions. For example, AS-path prepending can be included in the community attribute to instruct upstream domains to perform AS path prepending before sending route advertisements to their specific upstream domains [85][86].
- *NAT address translation*. This approach manipulates Network Address Translation (NAT) tables [87][88]. The NAT rules associate destination prefixes with the best ingress point such that the source address in packets for the destination is translated to the address of the chosen ingress point.
- *BGP Overlay*. An overlay policy control architecture (OPCA) has been proposed to separate the policy from routing so that a faster channel can be used to handle routing policy changes [89]. OPCA consists of several major components including policy agent and database, measurement infrastructure, message propagation, etc. The aims of OPCA are to solve the BGP convergence problem by improving route failover time and to balance the inbound traffic load for multi-homed domains.

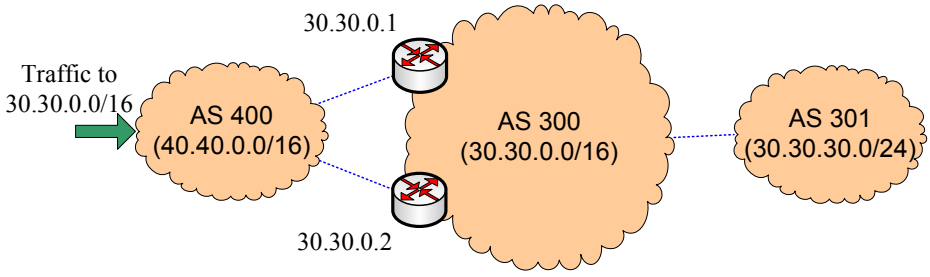

Figure 10 Inbound traffic engineering examples

4.4.2 Offline Inbound Traffic Engineering

In [90], the authors addressed an offline inbound inter-domain TE problem by optimizing AS-path prepending for stub domains. The problem is called Constrained Optimal Prepending (COP). The objective of COP is to determine the minimum number of prepended ASes for each prefix advertised through each ingress link such that the load constraint on each ingress link is satisfied. An essential assumption in this work is that the inbound route selection at the local domain is not affected by the setting of the local-pref attributes in its upstream domains. This is because, if local-pref is used, the upstream domains may send the traffic through another path towards the local domain using different ingress links. As a result, this makes the effect of AS-path prepending hard to predict. An Optimal Padding Vector (OPV) heuristic algorithm is proposed for solving the COP problem. The basic idea of the OPV algorithm is first to identify the most overloaded ingress link at each time, and then to increase the AS-path length by one of all customer prefixes to be advertised through the ingress link. The algorithm iterates until the traffic load received by each ingress link satisfies its maximum load constraint.

4.4.2 Online Inbound Traffic Engineering

In [82], the authors proposed a systematic and automated procedure named *AutoPrepend* to control inbound traffic using AS-path prepending. The basic operation of *AutoPrepend* is to artificially inflate the length of AS-path attribute in order to divert traffic onto different ingress links until the outcome network performance meets the traffic engineering goals. *AutoPrepend* is composed of four components:

(1) Passive measurement: To identify a set of top senders responsible for most of the inbound traffic.

(2) Active measurement: To send ICMP echo requests to the set of top senders and record the ingress links that receive the ICMP replies. A virtual *beacon prefix* with inflated AS-path length on one of the ingress links is sent to the set of top senders. The ingress links where the top senders respond to the beacon prefix are examined.

(3) Traffic prediction: Based on passive and active measurement, to predict the changes in the traffic volume on each ingress link when AS-path length increases. This is accomplished by comparing the measurements from the ICMP requests and the beacon prefixes described above.

(4) AS path update: To check if the predicted outcome satisfies the traffic engineering goals. If so, enforce the change by advertising the prefixes with the chosen AS-path length.

The authors in [83] proposed a greedy AS-path prepending heuristic algorithm to apply the abovementioned algorithm to the most heavily (or least) loaded ingress link and then to virtually inflate (or decrease) the AS-path length of the routes through the link by one until the TE goals are met.

In [88], the authors proposed the use of the NAT-based approach to control inbound traffic through the best ingress point. The instantaneous performance of the connected ingress points is continuously measured through active or passive measurement methods. The ingress link that gives the best performance is then selected for a given transfer.

A summary of the existing inbound TE work is presented in Table 8.

Reference	Optimization Objectives/metrics	TE Semantic	Implementation Techniques	Application Environment
[82]	Minimize link congestion and foresee performance impact	Online	AS path Prepending	Stub
[83]	Improve load balancing	Online	AS path prepending	Stub / Transit
[88]	Reduce Traffic request response time	Online	NAT	Stub
[90]	Minimize the number of prepending with the bandwidth constraint of ingress links	Offline	AS path prepending	Stub

Table 8. Inbound traffic engineering solutions

4.5 Cooperative Inter-domain Traffic Engineering

Since most domains in the Internet are self-governed entities and are effectively in competition with each other for customers, it is natural that they perform inter-domain TE individually without considering their neighbors. However, recent research has found that when adjacent domains perform their inter-domain TE selfishly, not only is the global network performance not optimized, but also the inter-domain TE strategies of each domain may adversely affect each other [91]. In this case, routing instability may occur, as domains need to change their path selection strategies whenever the TE decisions of their adjacent domains change. Such instability is primarily due to inter-domain TE policy conflicts between domains. A desirable way to achieve overall good TE performance is to encourage INPs to negotiate with each other in order to obtain a compromising solution that benefits them all. This is known as cooperative-based TE [92].

Cooperative-based TE relies on the negotiation between two adjacent domains to achieve an agreement on how traffic is routed between their networks. The TE objectives of the adjacent domains should be jointly considered in order to achieve a 'win-win' agreement that is satisfied by participating domains. Such an agreement can be determined through intelligent optimization methods, taking into consideration the topologies, TE objectives and traffic matrices of the two domains.

Compared to the existing effort on independent outbound and inbound TE, a very limited number of papers have investigated routing optimization using cooperative TE. In [93], the authors formulated an optimal peering problem for two domains that have agreed to establish peering relationships. The problem is to determine how many peering points are needed and how are they located such that the total cost of peering is minimized without compromising inter-domain service quality. With the peering point fixed, traffic is routed through the agreed ingress and egress points. A similar optimal peering problem has also been formulated in [94].

Apart from the optimal peering problem, distributed algorithmic mechanism design [95] has also been used for enabling cooperation between autonomous entities. In [96], the authors proposed a scheme in which individual domains disclose the real cost of routing within their networks. These costs are then used to compute lowest-cost routing solutions for all source and destination pairs so that social optimality is satisfied. The authors in [97] proposed using IP tunneling to establish explicit paths between source and destination domains through the

ingress links that are chosen to receive traffic. This approach is assumed valid in the environment where all network domains are cooperative. In addition, the authors in [98] proposed an algorithm for optimal route control among a group of cooperative multi-homed stub domains in order to reach a global TE solution that avoids oscillation caused by any conflict on TE objectives between domains.

5. Multicast Traffic Engineering

5.1 The Steiner Tree Problem

The problem of how to engineer optimally multicast traffic is far less well understood than unicast traffic engineering. A common objective of multicast traffic engineering is to minimize the total amount of bandwidth to be consumed. This objective is also known as *bandwidth conservation*, where conventional shortest path based routing paradigms are normally not optimal solutions. In the literature, bandwidth conservation in multicast routing is formulated as the directed Steiner tree problem, which has been proved to be NP-hard. The classic Steiner tree problem is described as follows. A network is represented with a graph $G = (V, E)$ with node set V and link set E. Each link $(i, j) \in E$ connects nodes i and j and has associated with it a metric of cost C_{ij}. There also exist a subset of nodes $D \subset V$, which corresponds to a set of multicast group members. The Steiner tree problem is to minimize the total cost of tree T that spans all the nodes in D, i.e.,

$$\text{Minimize} \sum_{(i,j)\in E} C_{ij} Y_{ij}, \quad i, j \in V$$

$$\text{Where } Y_{ij} = \begin{cases} 1 & if\ (i, j) \in T \\ 0 & otherwise \end{cases}$$

Research on this Steiner tree problem can be traced back to early 1980s. There exist two classic heuristics for this problem, namely the KMB algorithm [99] and the TM algorithm [100]. It is worth mentioning that the task of multicast traffic engineering is not necessarily identical to the classic Steiner tree problem. Apart from bandwidth conservation, there also exist some other TE objectives such as load balancing and maximizing throughput. Moreover, some other research on QoS-aware multicast routing also considers constraint-based Steiner tree problems such as delay [101] and delay variation [102]. These QoS-aware routing algorithms are not described in this paper, and interested readers can find an associated survey in [103].

5.2 MPLS Oriented Multicast Traffic Engineering

The most straightforward approach for MPLS based multicast traffic engineering is to set up point-to-multipoint (P2MP) LSPs, and this is where Steiner tree algorithms play a role. Before considering individual multicast TE schemes, we first investigate how to aggregate multicast traffic from different groups, which is an important procedure prior to LSP computation. This issue was first addressed in [104], and a scheme known as Aggregate Multicast was proposed. In this scheme, multiple multicast groups are forced to share one single P2MP LSP, even if the egress router set of these groups does not completely overlap. At the expense of some

extra bandwidth consumption, this approach is able to significantly reduce the total number of LSPs needed, thus improving scalability.

In [105], the authors proposed the Edge Router Multicasting (ERM) scheme for setting up P2MP LSPs only at the boundary of an MPLS domain. In ERM, multicast traffic aggregation in LSPs is confined to the network edge and thus the task is reduced to unicast TE within the domain. The authors studied two types of ERM: the first scheme is based on modifications to the existing multicast protocols while the second approach applies the Steiner tree based routing heuristic at edge routers.

Apart from an offline approach, online multicast traffic engineering has also been investigated, where future multicast sessions are not known *a priori*. In [106], Kodialam et al. extended their MPLS based online unicast TE scheme [42] to a multicast semantic. The basic objective is to accommodate as many multicast routing requests as possible without knowing about any incoming traffic in advance. The authors proposed a directed Steiner tree based online multicast routing algorithm for computing dynamic multicast trees with minimum bandwidth interference between individual sessions. [107] considered the dynamic multicast traffic engineering with both bandwidth and hop-count constraints, and they formulated this problem into Mixed Integer Programming (MIP). The objective of this work is to minimize the maximum link utilization as well as to satisfy the demand of hop-count constraint from individual multicast sessions.

5.3 IP Oriented Multicast Traffic Engineering

Despite its flexibility, explicit routing based TE approaches suffer from the complexity and cost associated with MPLS deployment. This problem becomes more serious in supporting multicast services, as P2MP (other than point-to-point) LSPs need to be maintained throughout the network. Compared to the unicast scenario, another difficulty in MPLS multicast traffic engineering is how to aggregate multicast flows, because different multicast sessions tend to have different egress routers attached with group members. As described above, this problem was addressed in the Aggregate Multicast scheme [104], but the associated scalability issue is still left open for further investigation. Naturally, one might wonder if it is also possible to engineering multicast traffic without MPLS enforcement, e.g., by using plain IP based paradigms? The answer is yes, but the number of relevant publications has been very small. The reason for this situation can be summarized as follows. First, Protocol Independent Multicast - Sparse Mode (PIM-SM) [108] uses the underlying IP unicast routing table for the construction of multicast trees, and hence it is difficult to decouple multicast traffic engineering from its unicast counterpart. Second, the enforcement of Steiner trees can be achieved through packet encapsulation and explicit routing mechanisms such as MPLS tunneling. However, this approach lacks support from hop-by-hop protocols, due to Reverse Path Forwarding (RPF) in the IP multicast routing protocol family. In PIM-SM, if multicast packets are not received on the shortest path through which unicast traffic is delivered back to the source, they are discarded so as to avoid traffic loops. Given the difference between the shortest path tree used by PIM-SM and the optimized minimum hop Steiner tree, engineered multicast traffic for bandwidth optimization through Steiner tree heuristics could result in RPF check failure.

The authors in [109] first stated that the theorem proved in [49] can also be applied to point-to-multipoint routes. This implies that a set of loop free Steiner trees can also be represented theoretically into shortest path trees with a proper set of link weights. Thus it is also possible to engineer multicast trees into Steiner trees for bandwidth conservation purposes without IP layer RPF checking failure. However, the authors did not propose how to achieve this type of tree representation in their work. To fill this gap, the authors of [110] proposed a genetic algorithm based approach to optimize PIM-SM multicast trees with bandwidth constraint by

setting properly the underlying IGP link weights. The objective is to achieve bandwidth conservation and load balancing through tuning the link weight of multi-topology enabled IGP (MT-IGP) protocols such as M-ISIS [111] and MT-OSPF [112]. The most distinct advantage of these two protocols is that they allow multiple sets of link weights for the same physical topology, with each corresponding to a specific type of traffic. In this scenario, multicast traffic engineering can be effectively decoupled from its unicast counterpart given the underlying MPLS-free environment. Figure 11 illustrates a simple example on how to conserve bandwidth in multicast routing by configuring optimized M-ISIS/MT-OSPF link weights. In this example, the single multicast source is node A, and nodes E, F, G are multicast group members. By conventional hop-count shortest path based PIM-SM routing, the bandwidth consumption is 6 units, with 1 unit consumed on each on-tree link. However, with proper link weight setting for MT-IGP, the optimal multicast tree for the same group is in effect a Steiner tree in terms of hop counts, with only 4 units of bandwidth being consumed (shown in figure (b)). In general, the practical approach is to optimize *multiple* multicast trees with only a set of MT-IGP link weights.

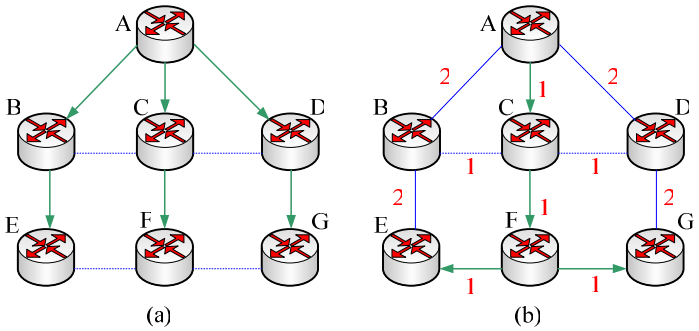

Figure 11. Steiner tree with IGP link weight optimization

6. Some Traffic Engineering Considerations

In this section, we discuss some important issues that need to be considered in routing optimization for advanced traffic engineering, specifically: TE robustness, TE interactions and interoperability between TE and overlay selfish routing.

6.1 TE Robustness

Most of the offline traffic engineering solutions described in this paper are based on the assumption that traffic matrices are accurate and the network is operating under normal conditions. However, to derive accurate traffic matrices is far from a trivial task due to the dynamic nature of Internet traffic. Moreover, failures, in particular logical ones, often occur in core networks. As a result, traffic fluctuation and network failure may cause the TE performance to be unpredictable, and thus make network management more complicated. Hence, it is necessary to make TE more robust in order to maintain the expected performance when any of those situations take place. Apart from achieving the expected performance, another advantage of this robust approach is that only one relatively stable network configuration is needed without frequent changes in response to the occurrence of any unexpected situation.

In the literature, robust TE has considered two issues: *link failure* and *traffic demand uncertainty*. The idea of the robust TE approach is first to model these issues as separate scenarios. For example, each link failure or traffic matrix represents a distinct scenario. Thereafter, a single TE configuration is produced that performs well at any given scenario.

As for the case of intra-domain link failure, which has been found to be common and transient [113], [114]-[117] proposed OSPF link weight setting algorithms to achieve the desired performance at any single link failure scenario. However, the computational complexity of algorithms increases significantly as the number of links in the network gets larger. In order to reduce such complexity, [114] further suggested performing robust TE optimization only on the *critical* links that have a significant impact on the overall network performance. For MPLS, the authors of [118] considered combined working and backup LSP optimization for all traffic demands. Specifically, a proactive ingress-to-egress restoration scheme with resource reservation was studied. The objective is to maximize the network's ability to carry future demands. Through this MPLS TE, the traffic carried over the network is fully restorable against all single event failures. Given that inter-domain peering link failures are as common and transient as intra-domain link failures, the authors of [119] proposed a local search heuristic to obtain an outbound inter-domain TE solution that is robust to any inter-domain link failure. Their objective is to minimize inter-domain link utilization both under normal state (no failure) and failure state with any single inter-domain link failure.

Traffic engineering in the case of multiple traffic matrix scenarios for the purpose of handling traffic demand uncertainty is relatively new. For intra-domain TE, Applegate and Cohen [120] found that it is possible to obtain a robust routing configuration that guarantees a nearly optimal utilization with a fairly limited knowledge of the applicable TMs. A similar work with link failure consideration was also proposed by the same authors [121]. Based on their work, the authors in [122] proposed algorithms to solve the robust intra-domain TE problem. Instead of using distinct traffic matrix scenarios, Mitra and Wang [123] proposed a stochastic optimization approach which assumes that the traffic demands are given probability distributions. Apart from being used for traffic matrix uncertainty, the robust TE approach can be used to obtain a high chance of performing well for multiple TMs, each of which represents traffic demands in a distinct period (e.g. days and evenings). This can be achieved through a set of OSPF link weight setting with the changing of a few link weights for different time periods [10]. This approach reduces the complexities in network management, as network operators do not need to change link weights on a regular basis. On the other hand, for inter-domain TE, the authors in [124] proposed an outbound TE approach based on scenario-based robust optimization, taking as input a set of inter-domain traffic matrices The objective of their work is to obtain an outbound TE solution that achieves good maximum inter-domain link utilization while minimizing the performance gap between the achieved solution and the optimal solution for any given inter-domain traffic matrix.

The ultimate objective of using robust TE approaches is to make network design and provisioning more predictable. This topic has been further receiving attention on designing a predictable Internet backbone network using novel approaches. Zhang and McKeown [125] propose using Valiant load-balancing over a fully-connected logical mesh for backbone network design. The aim of this approach is to achieve predictable and guaranteed performance, even when traffic matrices change and when links and routers fail. Kodialam et al. [126] propose a simple static routing scheme that is robust to extreme traffic fluctuations without requiring significant network over-provisioning.

6.2 TE Interactions

In Section 2 we classified traffic engineering into a set of categories. In this section we discuss TE interactions within each category from the viewpoint of routing optimization.

6.2.1 Intra-/Inter-domain TE Interaction

Much research has been conducted on intra-domain and inter-domain traffic engineering respectively, but how they work together as an integrated TE paradigm has not been well addressed. Recently, some publications have indicated that the interaction between intra- and inter-domain TE significantly impacts the overall performance [6]. First, any change of BGP ingress/egress point for traffic across a domain influences the intra-domain traffic matrix, and leads to significant impact on the effectiveness of intra-domain TE [6]. Hence, a more appropriate TE strategy is to take intra-domain conditions into consideration when performing inter-domain traffic engineering. For example, when selecting an egress point for any traffic trunk with bandwidth requirements, a prerequisite is to guarantee that at least one feasible intra-domain path with sufficient network resources exists between the ingress-egress pair. In [127], the authors proposed a joint optimization approach of intra- and inter-domain TE which is solved by a local search heuristic algorithm. Their results show that performing intra- and inter-domain TE simultaneously can maximize the network's capability to accommodate future traffic demands better than a sequential or nested approach that performs both TE separately.

The configuration of intra-domain TE can however also impact inter-domain path selection. A typical example is Hot Potato Routing (HPR) that has been often used by large INPs [7]. According to the BGP route selection policy, if multiple routes towards the same destination prefix are received through the same type of e/iBGP advertisement with identical values of local-preference, origin type, AS path length and MED, then the route having the lowest intra-domain IGP link weight is selected. Today, many INPs adopt HPR, which allows IGP link weights to influence egress router selection. By doing so, they hope that the traffic can be delivered out of the local domain using least number of hops (assuming each IGP link weight to be 1), which indicates that the least bandwidth resources are consumed. However, HPR also potentially leaves the inter-domain traffic instability problem in time of link failure. We reuse Figure 3 as an example. Assume that the INP of AS100 applies HPR for traffic delivery towards AS200 via egress node 10.10.10.1 according to his TE requirement. To achieve this, the configured IGP link weight for the shortest path between 10.10.10.3 and 10.10.10.1 (i.e., path A) should be lower than its counterpart between 10.10.10.3 and 10.10.10.2 (path C). Under this configuration, in case of a link failure on path A, the whole traffic trunk towards AS200 will shift automatically to use 10.10.10.2 as the egress point in AS100, if the IGP weight of the newly formed shortest path between 10.10.10.3 and 10.10.10.1 (e.g., path B) is larger than that of path C. In this scenario, not only does traffic routing within the network become unstable, but also the original TE objectives may be violated. With this example, we can see that intra-domain TE might also interact with inter-domain path selection. By showing the above examples, we indicate the importance of the intra-/inter-domain TE interaction, and we believe that further investigation in this area is worthwhile for more effective and robust TE.

6.2.2 MPLS/IP TE Interaction

In Section 2 we have shown respectively the distinct advantages and disadvantages of using IP/MPLS oriented traffic engineering schemes. Recently some proposals have been made to integrate IP and MPLS technologies to provide a hybrid TE solution. In [128], the authors suggested the option of using LSPs only to reroute the traffic trunks that contribute potentially

to network congestion, while the rest of the traffic is routed through plain IGP. In this case, the overhead introduced from LSP states can be reduced significantly at the expense of reasonably less flexibility in path selection. In the offline scenario, how to set up LSPs and configure IGP link weights so as to achieve overall network optimality is the key objective of the hybrid TE approach. If the IGP link weight is properly calculated then the number of LSPs needed for explicit routing to eliminate congestion can be reduced. In addition, hybrid online traffic engineering with both IGP and MPLS has also been investigated in [129]-[131]. These works aim at efficient allocation of unpredictable incoming traffic trunks onto different routing planes. In both cases, the interaction between IP oriented and MPLS oriented TE on top of the same physical network is of significant importance, as there exists a typical tradeoff between performance and scalability that should be taken into consideration by INPs.

6.2.3 Offline/Online TE Interaction

Despite the fundamental difference between offline/online TE that was described in Section 2, it is still possible, and even desirable in some circumstances, to combine them together for more sophisticated TE optimization. Although traffic matrices can sometimes be obtained in advance (e.g., through service level specifications) to provide the possibility of offline TE, it is not always the case that the overall traffic demands can be accurately predicted. In this case, static configuration according to the result from offline TE may not be able to handle unexpected traffic dynamics within each resource provisioning cycle. To compensate for this inefficiency, online traffic engineering can be used for dynamically adjusting traffic trunks according to the instant network condition obtained from real-time monitoring mechanisms. On the other hand, online traffic engineering should not discard completely the original configuration from offline TE, as significant traffic flapping and oscillation might be incurred, introducing network instability. In effect, a desired strategy to handle the relationship between offline and online TE is to allow offline traffic engineering to provide proper guidelines and restrictions to the online TE component, so that dynamic routing adjustment can be applied in a controlled manner. A typical example is the *TEQUILA* [14] architecture, where the offline network dimensioning (ND) functional block provides directives and non-specific “hard” values so as to leave space for unpredictable traffic fluctuations that will be handled by the Dynamic Route/Resource Management (DRtM, DRsM) functional blocks. In addition, a design-based routing has been proposed in [132] to use offline TE results to guide online traffic routing. Similarly, the BGP multi-paths mechanism also offers the functionality for the integration of offline/online inter-domain traffic engineering. During the offline network-provisioning phase, the INP may configure multiple routes towards a remote destination prefix, while BGP speakers can split traffic dynamically onto different next-hop peers based on the advertised inter-domain link bandwidth through eBGP [133].

6.2.4 Multi-plane TE Interaction

Finally, if we regard intra-/inter-domain TE interaction (including inter-domain TE itself) as a type of *horizontal* traffic engineering semantic between adjacent domains, then the terminology of *vertical* traffic engineering can be borrowed as the concept of network resource optimization across multiple network planes within a domain (Figure 12). Currently, there exist two major scenarios of traffic engineering with multiple network planes: (1) routing incongruence between different traffic types, e.g., IPv4/IPv6, unicast/multicast etc, and (2) different QoS requirements (e.g., DiffServ TE). Recently, with the advent of multi-topology aware routing protocols such as MT-OSPF, M-ISIS and MBGP [134], together with DiffServ-MPLS based solutions, vertical traffic engineering for multiple traffic types and QoS/TE requirements becomes a feasible option. However, even if these multi-plane routing protocols offer high flexibility in path selection, traffic engineering in the management plane concerning the *overall* resource optimization is still indispensable, as all types of traffic are mapped onto the same physical network infrastructure. In this case, traffic engineering for

individual network planes needs to be coordinated so as to achieve "vertical" optimization across all planes. Taking unicast/multicast TE as an example, the MT-IGP link weights can be assigned for unicast traffic and multicast traffic independently, aiming at different TE objectives (e.g., load balancing for unicast traffic and bandwidth conservation for multicast traffic). However, the calculation of link weights for the two planes should not proceed independently, as both unicast and multicast traffic are projected onto the same network resources. This means that the link weight setting for the two planes should concern overall TE optimization, other than the objectives in individual planes. It is also worth mentioning that multi-plane routing protocols are not absolutely necessary for routing of different traffic types. In fact, all types of traffic can be routed through a single plane with conventional OSPF/ISIS and BGP. In this scenario, configuration of the unique set of link weight and BGP path selection should include all TE objectives. Since multi-plane routing protocols have not been widely deployed in the Internet, it would be interesting to investigate the relevant performance against the scalability in Routing Information Base (RIB) that is needed to store the routing information for multiple planes, compared to the conventional single plane routing semantics.

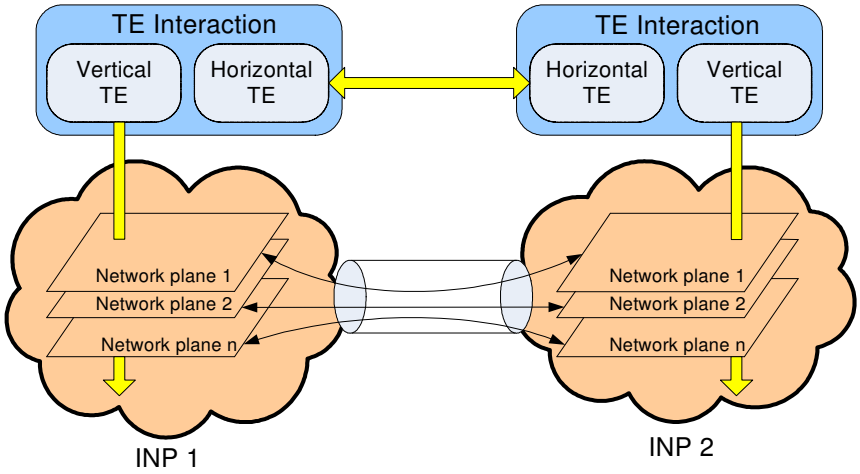

Figure 12 Horizontal/vertical TE interactions

6.3 Traffic Engineering vs. Overlay Selfish Routing

In some circumstances, there exist conflicts between TE objectives and end-to-end QoS demands from individual customers in which traffic engineering cannot satisfy the QoS requirements. In this case, overlay selfish routing is a flexible mechanism for end users to bypass TE constraints. A distinct characteristic of overlay routing is that path selection is performed by end hosts running applications according to their QoS requirements, and the underlying IP routing infrastructure is not aware of any overlay traffic[1]. In this sense, overlay routing is also known as selfish routing, as it does not consider the optimization for any other traffic within the network [135]. As it has been mentioned, TE aims at overall optimization of network performance by controlling traffic across the network. With the introduction of overlay routing, traffic engineering becomes less efficient because the routing of overlay traffic is outside the control of the INP. This problem has been identified recently and several research papers have addressed the interaction between TE and overlay routing. In [135], the

[1] This flexible functionality of overlay routing is very similar to MPLS explicitly routing. The key difference is that overlay routing is always performed by end users for their own QoS benefits, while MPLS explicit routing is normally adopted by INPs for TE purposes.

authors applied game theory to analyze the behavior of overlay routing and IP/MPLS oriented traffic engineering, taking end-to-end delay as a typical QoS metric. The result of their work showed that, through dedicated overlay routing, near optimal traffic delay can be achieved provided that the network layer routing of other traffic is static. However, network congestion still occurs at some hot spots within the network, because the overall traffic distribution cannot be fully managed by TE. Furthermore, the performance of IP oriented TE with overlay traffic coexistence was found to be very poor, while the situation can be improved using MPLS oriented traffic engineering with explicit routing and uneven splitting functionality. Other research work, such as [136], also indicated the same conclusion based on both theoretical and experimental analysis. As a conclusion, the more traffic in the network that is outside the management scope of the INP, the poorer the TE performance results. This indicates that excessive overlay traffic brings significant negative impacts to effective traffic engineering.

7. Summary

In this article we have provided an overview of routing optimization schemes for Internet traffic engineering. In order to systematically introduce various TE solutions in the literature, we classified them into a taxonomy according to four different criteria, namely intra-/inter-domain TE, MPLS/IP oriented TE, offline/online TE and unicast/multicast TE. Within each category, we specifically introduced classical TE solutions and also discussed corresponding advantages and disadvantages for each TE category. Moreover, we also foresee the importance of the interaction between complementary TE solutions within each category, and pointed out some insights for potential research topics. Finally, we addressed the relationship between TE and selfish overlay routing, both of which have been studied extensively and the importance of whose relationship has been recently realized.

References

[1] N. Hu et al, “Locating Internet Bottlenecks: Algorithms, Measurements and Implications”, Proc. ACM SIGCOMM 2004

[2] D. Awduche et al, “Overview and Principles of Internet Traffic Engineering”, RFC 3272, May 2002

[3] Y. Lee et al, “Traffic Engineering in Next-Generation Optical Networks”, IEEE Communications Surveys & Tutorials, 2nd Quarter, 2004, pp. 16-33

[4] D. Awduche et al, “Requirements on Traffic Engineering over MPLS”, RFC 2702, Jun. 1999

[5] O. Younis et al, “Constraint-based Routing in the Internet: Basic Principles and Recent Research”, IEEE Communications Surveys & Tutorials, 3rd Quarter, 2003, pp. 2-13

[6] S. Agarwal et al, “The Impact of BGP Dynamics on Intra-domain Traffic”, Proc. ACM. SIGMETRICS 2004

[7] R. Teixeira et al, “Network Sensitivity to Hot-Potato Disruptions”, Proc. ACM SIGCOMM 2004

[8] D. Awduche et al, “MPLS and traffic engineering in IP networks”, IEEE Communications Magazine, Vol. 37, Issue 12, pp. 42-47, Dec. 1999

[9] B. Fortz et al, “Internet Traffic Engineering by Optimising OSPF Weights”, *Proc. IEEE INFOCOM*, pp. 519-528, 2000

[10] B. Fortz et al, “Optimizing OSPF/IS-IS weights in a changing world”, IEEE JSAC. Vol 20, Issue 4, 2002, pp. 756-767

[11] B. Fortz et al, “Traffic engineering with traditional IP routing protocols”, IEEE Communication Magazine, Vol. 40, Issue 10, 2002, pp. 118-124

[12] B. Quoitin et al, "Interdomain traffic engineering with BGP", IEEE Communications Magazine, May 2003.

[13] R. Teixeira et al, “Dynamics of Hot-Potato Routing in IP Networks,” proc. ACM SIGMETRICS 2004.

[14] P. Trimintzios et al, “A Management and Contral Architecture for Providing IP Differentiated Services in MPLS-Based Networks”, IEEE Communication Magazine, May 2001, pp 80-88

[15] L. Andersson, et al, "LDP Specification", RFC 3036, Jan. 2001

[16] D. Awduche et al, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, Dec. 2001

[17] B. Jamoussi et al, "Constraint-Based LSP Setup using LDP", RFC 3212, Jan. 2002

[18] B. Braden et al, "Resource Reservation Protocol", RFC 2205, Sept. 1997

[19] D. Katz et al, "Traffic Engineering (TE) Extensions to OSPF Version 2", RFC 3630, Sept. 2003

[20] H. Smit et al, “Intermediate System to Intermediate System (IS-IS) Extensions for Traffic Engineering (TE)”, *RFC 3784*, Jun. 2004

[21] X. Xiao et al, “Traffic Engineering with MPLS in the Internet”, IEEE Network, Vol. 14, Issue 12, 2000, pp. 28-33

[22] D. Mitra and K.G. Ramakrishnan, “A Case Study of Multiservice, Multipriority Traffic Engineering Design for Data Networks”, Proc. IEEE GLOBECOM 99, pp. 1077-1083.

[23] Z. Wang et al, “Quality of Service Routing for Supporting Multimedia Applications”, IEEE JSAC Vo. 14, No. 7, 1996, pp. 1228-1234

[24] R. Guerin, et al, “QoS Routing Mechanisms and OSPF Extensions”, Proc. IEEE GLOBECOM 1997

[25] Y. Wang et al, “Explicit Routing Algorithms for Internet Traffic Engineering”, Proc. IEEE ICCCN’99, pp. 582-588

[26] Y. Lee et al, “A Constrained Multipaths Traffic Engineering Scheme for MPLS Networks”, Proc. IEEE ICC’02, pp. 2431-2436

[27] S. C. Erbas et al ”An Offline Traffic Engineering Model for MPLS Networks”, Proc. IEEE LCN’02, pp. 166-174

[28] M. Moh et al, “Supporting Differentiated Services with Per Class Traffic Engineering in MPLS”, Proc. IEEE ICCCN 2001, pp. 354-360

[29] N. Akar et al, “A Reordering-free Multipath Traffic Engineering Architecture for DiffServ-MPLS Networks”, proc. IEEE IPOM 2003, pp. 107-113

[30] R. Rabbat et al, “Traffic Engineering Algorithms Using MPLS for Service Differentiation”, Proc. IEEE. ICC’00, Vol. 2, pp. 791-795

[31] F. Le Faucheur et al, “Requirements for Support of Differentiated Services-aware MPLS Traffic Engineering”, RFC 3564, July 2003

[32] P. Trimintzios et al, “Quality of Service Provisioning through Traffic Engineering with Applicability to IP Based Production Networks”, Computer Communications, Vol. 26, 2003, pp. 845-860

[33] V. Tabatabaee et al, “Differentiated Traffic Engineering for QoS Provisioning”, Proc. IEEE INFOCOM 2005

[34] H. Saito et al, “Traffic Engineering Using Multiple Multipoint-to-point LSPs”, Proc. IEEE INFOCOM 2000

[35], G. Urvoy-Keller et al, “Traffic Engineering in a Multipoint-to-point Network”, IEEE JSAC Vol. 20, issue 4, pp. 834-849

[36] S. Bhatnagar et al, “Creating Multipoint-to-point LSPs for Traffic Engineering”, IEEE Communication Magazine, Jan. 2005, pp. 95-100

[37] P. Trimintzios et al, “Engineering the Multi-Service Internet: MPLS and IP-based Techniques”, Proc. IEEE ICT 2001

[38] A. Elwalid et al, “MATE: MPLS adaptive traffic engineering”, Proc. IEEE INFOCOM 2001

[39] S. Kandula et al, “TeXCP: Responsive Yet Stable Traffic Engineering”, Proc. ACM SIGCOMM 2005

[40] R. Boutaba et al, “DORA: Efficient Routing for MPLS Traffic Engineering”, Journal of network system management, Vol. 10, No. 3., 2002, pp. 309-325

[41] K.Kar et al, “Minimum Interference Routing of Bandwidth Guaranteed Tunnels with MPLS Traffic Engineering Applications”, IEEE JSAC Vol. 18, No. 12, 2000, pp. 2566-2579

[42] K. Kodialam et al “Minimum Interference Routing of Applications to MPLS Traffic Engineering”, Proc. IEEE Infocom 2000

[43] B. Wang et al, “A New Bandwidth Guaranteed Routing Algorithm for MPLS Traffic Engineering”, Proc. IEEE ICC’02

[44] C. Scoglio et al, “TEAM: A Traffic Engineering Automated Manager for DiffServ Based MPLS Networks”, IEEE Communication Magazine, October 2004, pp. 134 – 145

[45] P. Aukia et al, “RATES: A Server for MPLS Traffic Engineering”, IEEE Network, March/April, 2000, pp. 34-41

[46] J. C. de Oliveira et al, “SPeCRA: A Stochastic Performance Comparison Routing Algorithm for LSP setup in MPLS Networks”, Proc. IEEE Globecom’02, pp. 2190-2194

[47] J. C. de Oliveira et al, “New Preemption Policies for DiffServ Aware Traffic Engineering to Minimize Rerouting in MPLS Networks”, IEEE/ACM Trans. on Networking, Vol. 12, No. 4, pp. 733-745

[48] K. Walkowiak, “Survivable Online Routing for MPLS Traffic Engineering”, International Workshop QoFIS 2004, pp. 288-297

[49] Y. Wang et al, “Internet Traffic Engineering without Full Mesh Overlaying”, Proc. IEEE INFOCOM, Vol. 1, pp. 565-571, 2001

[50] G. Retvari et al, “On the Representability of Arbitrary Path Sets as Shortest Paths: Theory, Algorithms and Complexity”, *Proc. IFIP Networking*, pp. 1180-1191, 2004

[51] A. Feldmann et al, “NetScope: Traffic Engineering for IP Networks”, IEEE Network, March/April, 2000, pp. 11-19

[52] M. Ericsson et al, "A Genetic Algorithm for the Weight Setting Problem in OSPF Routing", Journal of Combinatorial Optimization, 6, pp. 299-333, 2002

[53] G. Retvari et al, “Practical OSPF traffic engineering”, IEEE Communication Letters, Vol. 8, Issue 11, 2004, pp.689-691

[54] A. Sridharan et al, "Achieving Near-Optimal Traffic Engineering Solutions for Current OSPF/IS-IS Networks", Proc. IEEE INFOCOM, pp. 1167-1177, Apr. 2003

[55] J. Wang et al, “Edge Based Traffic Engineering for OSPF Networks”, Computer Networks, Vol. 48 issue 4, pp. 605-625, 2005

[56] C. Labovitz et al., “Internet Routing Instability”, IEEE/ACM Transactions on Networking, Vol. 6, No. 5, October 1998.

[57] C. Villamizar, “OSPF Optimized Multipath (OSPF-OMP)”, IETF Internet-draft, draft-ietf-ospf-omp-02, February 1999.

[58] I. Gojmerac, T. Ziegler, F. Ricciato and P. Reichl, “Adaptive Multipath Routing for Dynamic Traffic Engineering”, Proc. IEEE Globecom, November 2003.

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

[60] L. Subramanian et al, “HLP: A Next Generation Inter-domain Routing Protocol”, Proc. ACM SIGCOMM 2005

[61] Cisco Systems, "BGP Multipath Load Sharing for Both eBGP and iBGP in an MPLS-VPN" 2005.

[62] L. Gao and J. Rexford, “Stable Internet routing without global coordination”, IEEE/ACM Trans. On Networking, Vol. 9, Issue 6., 2001, pp.681-692

[63] Y. R. Yang et al, “Stable Egress Route Selection for Interdomain Traffic Engineering,” to appear IEEE Networks Magazine, 2005.

[64] N. Feamster et al, "Guidelines for interdomain traffic engineering", ACM SIGCOMM Computer Communications Review, 33(5), 2003, pp. 19-30.

[65] J.P. Vasseur et al, "Inter-AS MPLS Traffic Engineering", Internet Draft, draft-vasseur-ccamp-inter-area-as-te-00.txt, Work in Progress, August, 2004.

[66] D. Goldenberg et al, "Optimizing Cost and Performance for Multihoming", Proc. ACM SIGCOMM Conference, 2004.

[67] S. Uhlig et al, "Interdomain Traffic Engineering with minimal BGP Configurations", Proc. of the 18th International Teletraffic Congress, Berlin, September 2003.

[68] S. Uhlig, "A multiple-objectives evolutionary perspective to interdomain traffic engineering in the Internet", to appear in the International Journal of Computational Intelligence and Applications, Special Issue on Nature-Inspired Approaches to Telecommunications, 2005.

[69] H. Wang, "Optimal ISP Subscription for Internet Multihoming: Algorithm Design and Implication Analysis", Proc. IEEE INFOCOM, 2005.

[70] K. Ho et al, "An Incentive-based Quality of Service Aware Algorithm for Offline Inter-AS Traffic Engineering", Proc. IEEE IPOM, 2004.

[71] T.C. Bressoud et al, “Optimal Configuration for BGP Route Selection,” proc. IEEE INFOCOM, 2003.

[72] T.W. Chim and K.L. Yeung, "Time-Efficient Algorithms for BGP Route Configuration", Proc. IEEE ICC, 2004.

[73] K. Ho et al, "On Egress Router Selection for Inter-domain Traffic with Bandwidth Guarantees", Proc. IEEE HPSR, 2004.

[74] K. Ho et al, "Multi-objective Egress Router Selection Policies for Inter-domain Traffic with Bandwidth Guarantees", Proc. IFIP Networking Conference, 2004.

[75] S. Uhlig and B. Quoitin, "Tweak-it: BGP-based interdomain traffic engineering for transit ASes", Proc. EuroNGI Conference, 2005.

[76] M. Howarth et al., “Provisioning for Inter-domain Quality of Service: the MESCAL Approach,” IEEE Communications Magazine, Vol. 43, No. 6, June 2005, pp. 129-137.

[77] M. Howarth et al., “End-to-end Quality of Service Provisioning Through Inter-provider Traffic Engineering,” Computer Communications, Vol. 29, No. 6, March 2006, pp. 683-702.

[78] M.R. Gary and D.S. Johnson, “*Computers and Intractability: A Guide to the Theory of NP-Completeness*”, W.H. Freeman and Company, San Francisco, CA, 1979

[79] S. Uhlig and O. Bonaventure, "Designing BGP-based outbound traffic engineering techniques for stub ASes", ACM SIGCOMM Computer Communications Review, October 2004.

[80] A. Akella et al, "A Measurement-Based Analysis of Multihoming", Proc. ACM SIGCOMM Conference, 2003.

[81] S. Lee et al, "Exploiting AS Hierarchy for Scalable Route Selection in Multi-Homed Stub Networks", Proc. ACM IMC, 2004.

[82] R.K.C. Chang and M. Lo, "Inbound Traffic Engineering for Multihomed Ass Using AS Path Prepending", IEEE Network Magazine, March/April 2005.

[83] H. Wang et al, "Characterizing the Performance and stability Issues of the AS Path Prepending Method: Taxonomy, Measurement Study and Analysis", ACM SIGCOMM ASIA WORKSHOP, 2005.

[84] B. Quoitin et al, "A performance evaluation of BGP-based traffic engineering", International Journal of Network Management, 15(3), May-June 2005.

[85] S.R. Sangli et al, "BGP Extended Communities Attribute", Internet Draft, draft-ietf-idr-bgp-ext-communities-08.txt, February 2005.

[86] B. Quoitin et al, "Interdomain Traffic Engineering with Redistributed Communities", Computer Communications, October 2003.

[87] S. Kalyanaraman, "Load Balancing in BGP Environments using Online Simulation and Dynamic NAT", Presented at the Internet Statistic and Metrics Analysis Workshops, 2001.

[88] A. Akella et al, "Multihoming Performance Benefits: An Experimental Evaluation of Practical Enterprise Strategies", Proc. USENIX Conference, 2004.

[89] S. Agarwal et al, "OPCA: Robust Interdomain Policy Routing and Traffic Control", Proc. IEEE OPENARCH, 2003.

[90] R. Gao et al, "Interdomain Ingress Traffic Engineering through Optimized AS-Path Prepending", Proc. IFIP Networking Conference, 2005.

[91] R. Mahajan et al, "Towards Coordinated Interdomain Traffic Engineering", Proc. ACM HotNets-III Workshop, 2004.

[92] R. Mahajan et al, "Negotiation-based Routing Between Neighboring Domains", Proc. Networked Systems Design and Implementation (NSDI), May 2005.

[93] D. Awduche et al, "An Approach to Optimal Peering Between Autonomous Systems in the Internet", Proc. IEEE ICCCN, 1998.

[94] R. Johari and J.N. Tsitsiklis, "Routing and peering in a competitive Internet", Technical Report P-2570, MIT Laboratory for Information and Decision Systems, January 2003.

[95] J. Feigenbaum et al, "Distributed algorithmic mechanism design: Recent results and future directions", Proc. of the 6th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, September 2002.

[96] J. Feigenbaum et al, "A BGP-based Mechanism for Lowest-Cost Routing", Proc. of the Annual ACM Symposium on Principles of Distributed Computing, 2002.

[97] B. Quoitin and O. Bonaventure, "A Cooperative Approach to Inter-domain Traffic Engineering", Proc. EuroNGI, 2005.

[98] Y. Liu and N. Reddy, “Route Optimization among a Group of Multihomed Stub Networks”, Proc. IEEE GLOBECOM 2005.

[99] L. Kou et al, "A Fast Algorithm for Steiner Trees", Acta Informatica, 15, pp. 141-145, 1981

[100] H. Takahashi et al, “An Approximate Solution for the Steiner Problem in Graphs”, Math. Japonica 6, pp533-577

[101] Q. Zhu et al, “A Source Based Algorithm for Delay-constrained Minimum-cost Multicasting”, Proc. on IEEE INFOCOM, Vol. 1, pp. 377-385, 1995

[102] G. N. Rouskas et al, "Multicast Routing with End-to-end Delay and Delay Variation Constraints", IEEE JSAC Vol. 15, No. 3, pp. 346-356, Apr. 1997

[103] B. Wang, J. C. Hou, “Multicast Routing and Its QoS Extension: Problems, Algorithms and Protocols”, *IEEE Network*, pp. 22-36, Jan./Feb. 2000

[104] A. Fei at al, “Aggregated Multicast with Inter-Group Tree Sharing”, Proc. International Workshop on Networked Group Communications (NGC) 2001

[105] B. Yang et al, “Multicasting in MPLS Domains”, Computer Communications, 27(2), 2004, pp. 162-170

[106] M. Kodialam et al, “Online Multicast Routing with Bandwidth Guarantees: A new Approach Using Multicast Network Flow”, IEEE/ACM Trans. on Networking, Vol. 11, No. 4, pp. 676-686, 2003

[107] Y. Seok et al, “Explicit multicast routing algorithms for constrained traffic engineering”, *Proc. IEEE ISCC*, pp. 455-461, 2002

[108] B. Fenner, “Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)”, draft-ietf-pim-sm-v2-new-11.txt, April 2005, work in progress

[109] Y. D. Meisel et al, “Multicast Routing with Traffic Engineering: a Multi-Objective Optimization Scheme and a Polynomial Shortest Path Tree Algorithm with Load Balancing”, Proc. CCIO 2004

[110] N. Wang et al, “Bandwidth Constrained IP Multicast Traffic Engineering Without MPLS Overlay”, Proc. IEEE/IFIP MMNS, 2004

[111] T. Przygienda et al, “M-ISIS: Multi Topology (MT) Routing in IS-IS”, Internet Draft, draft-ietf-isis-wg-multi-topology-09.txt, March. 2005, work in progress

[112] P. Psenak et al, “Multi-Topology (MT) Routing in OSPF” Internet Draft, draft-ietf-ospf-mt-03.txt Mar. 2005, work in progress

[113] G. Iannaccone et al., “Analysis of Link Failures in an IP Backbone”, Proc. ACM IMW 2002.

[114] B. Fortz et al, “Robust Optimization of OSPF/IS-IS Weights”, Proc. INOC, pp. 225-230, 2003

[115] A. Nucci et al, “IGP Link Weight Assignment for Transient Link Failures”, Proc. International Teletraffic Congress, August 2003

[116] A. Sridharan et al, “Making IGP Routing Robust to Link Failures”, Proc. Networking 2005.

[117] D. Yuan, “A Bi-Criteria Optimization Approach for Robust OSPF Routing,” proc. IEEE IPOM 2003, pp. 91-97.

[118] E. Karasan et al, “Robust Path Design Algorithms for Traffic Engineering with Restoration in MPLS Networks”, IEICE Transactions on Communication, Vol. E86-b, No. 5, pp 1632-1640

[119] M. Amin et al., “Making Outbound Route Selection Robust to Egress Point Failure”, Proc. IFIP Networking 2006.

[120] D. Applegate and E. Cohen, “Making Intra-Domain Routing Robust to Changing and Uncertain Traffic Demands: Understand Fundamental Tradeoffs,” proc. *ACM SIGCOMM* 2003.

[121] D. Applegate et al., “Coping with network failures: routing strategies for optimal demand oblivious restoration,” proc. *ACM SIGMETRICS* 2004.

[122] C. Zhang et al., “On Optimal Routing with Multiple Traffic Matrices,” *IEEE INFOCOM* 2005.

[123] D. Mitra and Q. Wang, “Stochastic Traffic Engineering for Demand Uncertainty and Risk-Aware Network Revenue Management,” *IEEE/ACM Transactions on Networking*, vol. 13, no. 2, April 2005,

[124] K.H. Ho et al., “A Robustness Approach to Inter-AS Outbound Traffic Engineering”, Proc. IEEE ICC 2006.

[125] R. Zhang and N. Mckeown, “Designing a Predictable Internet Backbone Network,” proc. ACM HotNets 2004.

[126] M. Kodialam, T.V. Lakshmann and S. Sengupta, “Efficient and Robust Routing of Highly Variable Traffic,” proc. ACM HotNets 2004.

[127] K.H. Ho et al., “Joint Optimization of Intra- and Inter-Autonomous System Traffic Engineering”, Proc. IEEE/IFIP Network Operations and Management (NOMS), April 2006.

[128] J. Boyle et al, “Applicability Statement for Traffic Engineering with MPLS”, RFC 3346, August, 2002

[129] H. Pham et al, “Hybrid Routing for Scalable IP/MPLS Traffic engineering”, Proc. IEEE ICC’ 2003, Vol. 1 pp. 332-337

[130] A. Bagula, “Online Traffic Engineering: A Hybrid IGP + MPLS Routing Approach”, Proc. International Workshop of QoFIS 2004

[131] A. Bagula, “Hybrid IGP + MPLS Routing in Next Generation IP Networks: An online Traffic Engineering Model”, Proc. International Workshop QoSIP 2005

[132] A. Elwalid, “Routing and Protection in GMPLS Networks: From Shortest Paths to Optimized Designs”, Journal of Lightwave Technology, vol. 21, no. 11, pp. 2828-2838, Nov 2003.

[133] “BGP Bandwidth Link”, online Cisco white paper, http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122newft/122t/122t2/ftbgplb.htm

[134] T. Bates et al, “Multiprotocol extensions for BGP-4”, RFC 2858, June 2000

[135] L. Qiu et al, “On Selfish Routing in Internet-like Environments”, Proc. ACM SIGCOMM 2003, pp. 151-162.

[136] Y. Liu et al, “On the Interaction Between Overlay Routing and Traffic Engineering”, Proc. IEEE INFOCOM 2005.