Traffic Engineering towards the Assurance of Quality in IP networks: Trends and Perspectives

P. Fafali[•] Y. Despotopoulos[|] Ch. Patrikakis[+], N. Minogiannis[*]

Abstract

The need for establishing bandwidth guaranteed paths in IP networks and the requirement for making optimal use of the available resources becomes more and more crucial due to the significant development of data-intensive multimedia applications. In this paper, we discuss the techniques and the mechanisms for exercising traffic engineering in contemporary IP networks under the prism of exploiting historical monitoring information collected from the operational environment. Current state of the art and leading directions in the area of traffic engineering are presented in relation to the architectures and protocols that lay the foundation for building the commercial Internet. Issues that can lend insight into the route determination process such as the type of data to be monitored, are discussed along with the difficulties and limitations encountered in obtaining a traffic matrix. Our claims are substantiated through a set of simulation experiments conducted. In conclusion, we provide some directives on the deployment of a history aware traffic engineering mechanism, and report on issues that need to be taken into consideration.

Keywords*:* *Traffic engineering, history monitoring information, traffic matrix, QoS.*

1 Introduction

As Internet evolves into a standard communications network with emerging real time applications, such as voice and video, new techniques must be introduced for the management of the available assets. This need is enhanced by the fact that priceless network resources are not abundant. At one extreme, customers are willing to pay enough in order to enjoy the high quality services they desire. At the other extreme, ISPs, in their effort to overpower competitors, aim at utilizing their network infrastructures in a more economical way so as to increase the potential for accommodating future demands. To meet both users' and ISPs requirements, a balance between these two extremities must be reached. That is where the impetus for enforcing Traffic Engineering (TE) lies.

The fundamental goal of Traffic Engineering is to run the network efficiently at both traffic and resource levels. It is the framework that deals with the issue of performance evaluation and optimization. Thus, traffic engineering implies several ranges of objectives, including cost minimization, congestion avoidance, load balancing, multi-path routing, preempting and network survivability. Basically, traffic demands, collected requirements, imposed constraints

---

[•] Telecommunications Lab, NTUA, Heroon Politechniou 9, Greece 15773, pfafali@telecom.ntua.gr
[|] Telecommunications Lab, NTUA, Heroon Politechniou 9, Greece 15773, ydes@telecom.ntua.gr
[+] Telecommunications Lab, NTUA, Heroon Politechniou 9, Greece 15773, bpatr@telecom.ntua.gr
[*] Telecommunications Lab, NTUA, Heroon Politechniou 9, Greece 15773, minogian@telecom.ntua.gr

and performance objectives are used as input to the traffic engineering system in order to direct the network into the desirable operating point by means of various functions. In [2], the basic components of Internet traffic engineering are illustrated. These components are bound into an interactive model that comprises the following phases:

- Measurement phase.
- Modeling and Analysis phase.
- Performance optimization phase.

Recent work emphasizes the need to incorporate monitoring information in making routing decisions. In this context, we will briefly discuss only the measurement phase. During this phase, statistics are collected from the operational network. These metrics are chosen to capture attributes such as traffic patterns, traffic trends and link utilization. The measurement collection cycle is considered very critical to the traffic engineering process as it acts as a feedback mechanism to the control subsystems. It is also important for the evaluation of the effectiveness of the traffic engineering framework during the analysis step. Moreover, monitored data can help meet the optimization objective set if it is collected in a systematic basis.

The main motivation of this paper is to shed light on the new trends that steer the traffic engineering mechanisms. A common attribute amongst the topics tackled in this area of research is that they assume the existence of a traffic matrix used to pre-program routing plans or to estimate network parameters. The point-to-point traffic matrix is obtained from measurements retrieved from the operational environment. This paper surveys existent schemes for exercising traffic engineering in IP networks (connectionless and connection-oriented) that count on traffic information gathered from the real network and discusses the individual goals set by the proposed solutions. The inherent limitations of these schemes will be also pointed out. In addition, the methods used for the construction of the traffic matrices are presented and future directions for their integration in the TE framework are identified.

The rest of the paper is organized as follows. Section 2 introduces the two prevailing traffic engineering approaches in terms of the underlying network architectures that enable its deployment: the connectionless and the connection-oriented. Their pros and cons are thoroughly analyzed and the most representative solutions in each category are reviewed. Section 3 is devoted to discussing how the traffic engineering framework can be ameliorated by the use of history monitoring information. The model upon which the emerging routing algorithms are built is demonstrated. Furthermore, this section deals with the issues pertaining to the formulation and use of the traffic matrix which is the basic building block of the new traffic engineering paradigm. In Section 4, we present simulation tests performed for a representative scheme in the area of traffic-aware routing. Finally, Section 5 concludes the paper by providing directives on the deployment of history aware TE mechanisms and highlighting issues for further study.

2 Connectionless vs. connection-oriented Traffic Engineering

Two broad taxonomies dominate the debate for exercising traffic engineering in the Internet. These are the connectionless or IP-based approach and the connection-oriented or signaling approach. The latter was actually introduced in order to remedy the shortcomings of the former. In the following sections those two schemes are analytically discussed.

2.1 Connectionless Traffic Engineering

Connectionless traffic engineering model counts on traditional Interior Gateway Protocols (IGP) such as OSPF (Open Shortest Path First) [4] and IS-IS (Intermediate System-Intermediate system) [5]. OSPF and IS-IS are link state protocols based on a shortest path algorithm. They develop and maintain full knowledge of network routers, as well as how they interconnect. This is achieved via the exchange of Link-State Advertisements (LSAs) which are flooded by routers in order to construct their topological databases. Depending on the link states, the algorithm produces a shortest path tree to all destinations based on the static weights assigned to the links. The link weights are initially configured by network administrators. Cisco suggests as default weights the inverse of the interface available bandwidth in the sense that it helps traffic to circumvent likelihood “hot spots” [3].

The problem of mapping traffic onto physical links in connectionless TE approach boils down to the use of the least-cost routes. The desired result is obtained through the selection of the appropriate link weights. Most of the difficulties induced in connectionless scheme are related to the destination-based forwarding paradigm. Data-plane (packet forwarding) mechanism and control plane (routing) functions are coupled. Owing to that, changes in one mechanism mandate changes to the other too, meaning that software and hardware has to be upgraded, which is overly restrictive for large-scale Autonomous Systems (ASs). Moreover, traffic source has no control over the path selection procedure and by no means can it influence the latter. Consider the case where due to some policy/ administrative constraint or network deficiency (e.g. a specified link is out of operation or experiences congestion) the selected path should avoid certain links. But the least-cost algorithm that resolves the way incoming traffic flows into the network is inclined to selecting the shortest paths even if these are overloaded, while longer and underutilized paths may be available. Thus, load balancing which is a fundamental goal of traffic engineering cannot be met. This holds true even in the case of Equal-Cost Multipath (ECMP). Specifically, although ECMP allows traffic to be carried along multiple paths, the forwarding mechanism performs load balancing across these paths by equally splitting traffic among shortest paths with the same cost on the respective set of next hops.

Despite the aforementioned limitations, it is still a fact that OSPF and IS-IS routing protocols prevail across LANs. Furthermore, many extensions for improving their inefficiencies have been proposed. Their scalability and ease of deployment have made them widely expanded. The research efforts that have been contributed to the area of connectionless traffic engineering are the proof of concept. These approaches try to accomplish traffic engineering indirectly by affecting link weights. Note that finding optimal link weights for a set of projected traffic demands under constraints is proved to be NP-hard problem [7].

In one of the earlier approaches [7], a local search algorithm has been developed in the context of a fixed network topology and known demand matrix. Under this scheme, authors estimate optimum OSPF link weights that support more demands than the Cisco’s default weights and lead to performance close to that of optimal general routing. However, changing the link weights entails a transient period when the routers flood LSAs in order to update the calculated weights as well as their routing tables with new shortest paths. The migratory status that the network undergoes during the convergence results in routing loops and of out-of-order arrival of TCP packets.

Frequent change of link weights is not the best solution. For this reason, in [8], a heuristic is proposed in order to achieve improved performance through changing as few link weights as possible. It must be also noted that there is a trade-off between optimality and the number and the frequency of link weight changes. Inspired by [7], several approaches have been proposed. In [9], a genetic algorithm for solving the OSPF weight setting problem is demonstrated. Similarly, the optimization objective set in [10] yields optimal performance in terms of packet loss rate by dynamic optimization of OSPF weights. This scheme outperforms the local search approach adopted in [7] regarding the number of iterations needed to obtain a “good” link weight setting.

As it can be inferred from the references cited above, the main focus is concentrated on formulating optimization problems based on an estimated traffic matrix. In the mathematical models adopted, linear or non-linear, the bandwidth values appearing in the traffic matrices are expressed as a function of cost, delay or packet loss by the use of queueing models. By forcing the selected quality parameter to be minimized, these schemes lead to the calculation of optimal link weights. Solving that kind of problems is not an easy task and on this purpose, local search combinatorial, genetic, memetic algorithms or hybrid approaches are usually chosen. A criterion that determines the selection among these alternatives is the computational complexity required for achieving an optimal solution. That is of crucial importance especially in cases that solutions obtained must immediately feed a dynamic traffic engineering process such as the online simulation framework used in [10]. It should be noted that in conditions where time is scarce, the trade-off between the time required to approximate a solution and the optimality achieved should be weighted with respect to the performance improvement gained.

2.1.1 Connection-oriented Traffic Engineering

The connection–oriented approach of Traffic Engineering refers basically to Multi-Protocol Label Switching (MPLS). MPLS has been actually proposed as a means to alleviate the shortcomings of the traditional, unconstrained interior gateway protocols. Actually, traffic engineering is regarded as one of the most significant reasons for MPLS [11]. MPLS supports explicit route set-up between a source (ingress point) and a destination (egress point) and hence, the ability to distribute traffic into the network efficiently and keep the latter in a well-balanced state. In MPLS, packets passing through an ingress router are encapsulated with labels that are then used to forward packets along the Label Switched Paths (LSPs). The assignment of labels to packets is based on the concept of Forwarding Equivalence Class (FEC). The classification into FECs is done based on the information carried in the IP header of the packets and the local routing information maintained by the ingress router.

The strong point of MPLS is the clean separation between control plane and data plane. MPLS control plane is responsible for establishing LSPs while the data plane (forwarding) performs label swapping operations. This de-coupling between forwarding and control plane allows the introduction of new services by simply changing the way that packets are mapped to LSPs. It also supports overlay capabilities along with the advantages that the overlay model has. Traffic engineering with MPLS requires the combination of constrained based routing and the use of enhanced IGP link states [11]. In contrast to IGP protocols, MPLS allows the enforcement of administrative constraints. A significant advantage of MPLS is the ability to collect LSP statistics that can be used for the construction of traffic matrix. In addition, it facilitates load balancing through traffic splitting based on specified load ratios and it also enables path re-routing. Another feature is the ability to map Differentiated Services (DiffServ) classes to LSPs in order to enable multi-service functionality. All the above-mentioned features of MPLS make it a powerful traffic engineering solution. However, MPLS has also problems. Specifically, the signaling mechanism used to establish LSPs burdens the network with additional load and complexity. Lastly, other problems regard the triple mapping of: incoming traffic to FECs, FECs to LSPs and LSPs onto the physical topology [11].

MPLS-based traffic engineering has been extensively researched in order to provide solutions that facilitate traffic movement through the network. The volume of the work devoted to the specified area confines our reference to the findings that are based on long-term traffic measurements. Policy-Based Routing (PBR) [6] refers to an interesting approach aware of the total expected bandwidth for all source-destination pairs, so called as traffic profiles, used in a pre-processing phase. PBR is aimed at pre-allocating link resources to aggregates of requests based on the multicommodity network flow problem whose objective is to maximize the amount of flow sent through the network. Therefore, towards this end, it is prone to splitting traffic over multiple paths between a source-destination pair. So, though it pays off concerning load balance and network utilization, it proves insufficient in routing high bandwidth demands.

The results presented in [6] demonstrate the effectiveness of the proposed algorithm, and prove that offline processing of monitored data is crucial in order to establish the guidelines for accommodating incoming traffic through advanced routing control. Its advantage lies in the fact that pre-provisioning of resources is allowed while the processing cost is moved from the online to the offline phase. Therefore it appears promising for implementing more sophisticated algorithms since time and computational complexity are not constraints for an offline procedure. Routing has substantial influence on key performance issues of networks such as congestion, throughput, delay, and resource utilization. Owing to that it comprises an area we should further delve into and correlate with historical information.

3 History monitoring information featuring QoS mechanisms

Though traffic demands cannot be easily predicted and the chances that yesterday network traffic will be precisely repeated in today's scenario are few, macroscopically, traffic profile in communication networks depends on the period of the day. One reason for this dependency is that most of the traffic carried during the day is professional while residential traffic dominates in the evening [1]. Consequently, we can deal with more predictable periodic changes, simply by thinking of two independent routing problems: one for the morning and one for the evening. Also, the most obvious characteristics of Internet traffic are the strong diurnal and weekly cycles as well as long-term trends [21].

3.1.1 History aware TE scheme

Bandwidth is a key resource to manage in communication networks. It provides meaningful indications about the traffic that passed through a network. Along with control functionality, history monitoring information can provide the benchmark to decide on the technique applied for mapping traffic to the network installed capacities and on the routing algorithms to be employed. This means that traffic measurements collected from previous routing scenarios and topology data gathered from the operational network can be of determinant importance in provisioning what will be the best way of allocating available resources in the near future. Also, according to the classification made in [2], there is a category of traffic engineering systems called time-dependent systems. In time-dependent traffic engineering, historical information based on periodic variations in traffic is used to pre-program routing plans and other TE control mechanisms. Time dependent algorithms do not try to respond to random traffic variations or network dynamics. However, though the majority of traffic engineering aspects has received much attention, the concept of integrating historical information into the TE framework has not been systematically investigated.

Generally, the history information process model can be represented of the following steps:

- **Step 1**: The initial step required is to clarify the exact measurements that will be meaningful to building the TE scheme. These measurements are selected so as to capture the historical information and feed the online routing algorithm.
- **Step 2**: In this step measurements are collected. The periods that exhibit the same weekly, daily or even holiday cycles regarding the amount of traffic that passes through the network should be approximated.
- **Step 3**: Traffic matrix is generated in order to provide the policies and strategies that will be enforced in routing the incoming traffic. This is namely the offline phase of TE.
- **Step 4**: Routing requests that arrive online are served based on the rules formulated during the pre-processing phase.

The history aware TE model is depicted in Figure 1.

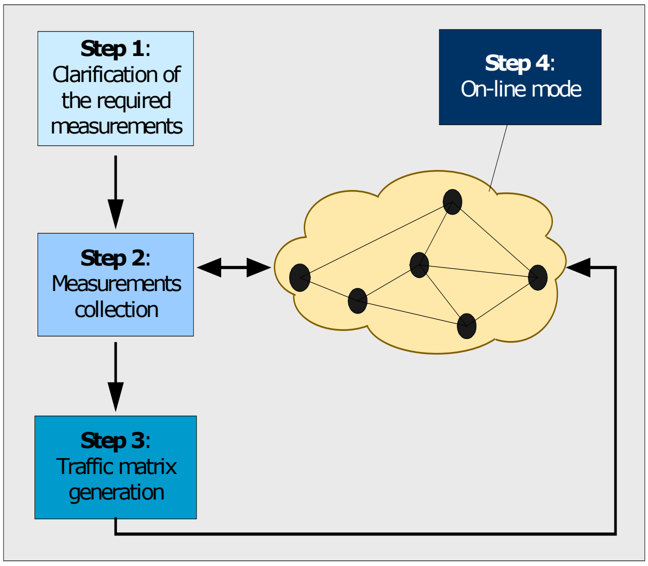

Figure 1. History aware TE scheme

3.1.2 Traffic Matrix Generation

A traffic matrix is the representation of the traffic characteristics between an abstract set of ingress and egress points in a communication network. Traffic matrices present a view of the network state in the past and the information they provide is a major tool for a wide variety of traffic engineering tasks including load balancing, routing protocols configuration, provisioning, capacity management, and fault recovery. Depending on the desired task to be performed, network operators gather statistics from the network in order to populate traffic matrices. The latter may be categorized based on the type of data they carry, the spatial criteria used for their creation, the level of aggregation they refer to, the temporal granularity they have, and the methodology used for their construction.

The typical metric populated in a traffic matrix is the traffic volume exchanged between a source-destination pair. However, this is not restrictive since other QoS parameters such as delay or packet loss are also important for carrying out traffic engineering tasks. Network operators need extra metrics to help them manage their networks. For example, several SLAs are established with specific performance characteristics such as delay constraints, especially when they refer to VoIP applications. High packet loss connotes congestion or link failure.

Another classification is based on the spatial representation of the network traffic over the network paths. In [13], the authors distinguish three matrix types, namely path matrix, traffic matrix and demand matrix. The path matrix representation specifies the data flow between every path in every source-destination pair. The traffic matrix specifies the traffic volume exchanged between source-destination pairs independently of the path followed. The last type is the demand matrix, which represents the traffic demand between every ingress point to a set of egress points of the network.

The aforementioned categories classify the data types and the spatial boundaries of a traffic matrix. In [12], the level of traffic aggregation is introduced as another classification criterion for traffic matrices. Depending on the location of the source and destination endpoints, the traffic matrix may comprise different levels of data aggregation. These levels are presented in Figure 2.

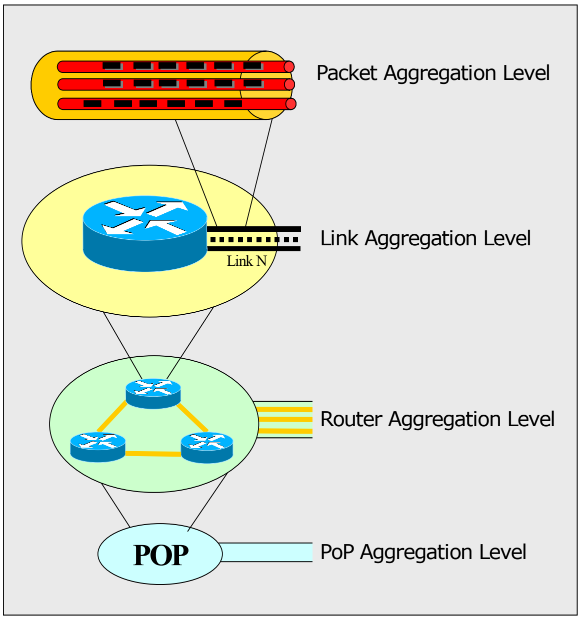

Figure 2. Levels of traffic aggregation

According to the level of traffic aggregation, the following categories are discriminated: the address prefix level, the link level, the router level and the POP level. The aggregation level is very critical because as statistics granularity increases, it is more complicated to gather them.

Depending on the monitoring period reflected in the traffic matrix we may also have small, medium and large time scale classification. Traffic matrices in different time scales serve diverse traffic engineering functions. In cases where some parameters have to be calculated in macroscopic scale, the volume of data to be stored and processed is increased. Small time scales (minutes to hours) are useful for diagnosis of congestion, router failures and Denial of Service attacks, medium (hours to weeks) are functional for routing table updates and introduction of new customers and peers. Finally, long time scale traffic matrices are used for capacity planning, evaluation of network design and new protocol deployment.

The construction of traffic matrices is based on direct measurements on network endpoints, both passive and active. Passive measurements include data collection by using the widely deployable by network element manufacturers SNMP protocol, or by applying the tcpdump tool [16], which sets a network interface to promiscuous mode on the desired links. Active methods imply injecting probe packets into the network to gather statistical data along the packets trip. Using only direct measurements to generate traffic matrices is extremely expensive in terms of CPU power consumed on the network elements, storage requirements and processing of statistics. Also, direct flow-level measurements require additional infrastructure support. At finer levels of detail, i.e. packet level, the previously mentioned techniques cannot provide accurate data especially for high speed backbone links. In a large ISP network where the total number of source-destination pairs is much larger than the number of links, SNMP measurements are prohibitive.

To overcome the limitations of traffic matrix generation from direct flow-level measurements, several techniques have been proposed in the literature [14]. These approaches generally try to infer the traffic matrix using data acquired from the network and applying statistical techniques. These techniques use different approaches such as gravity modeling, linear programming, bayesian estimation, network tomography and maximum likelihood estimation. The information used in TM inference techniques is basically link load data from

SNMP, routing information (link weights), topological information (peerings, access links) and assumption on the distribution of demands.

4 Simulation environment and experiments

The purpose of this section is to illustrate how effective traffic engineering can be when routing parameters are set based on a traffic matrix. In the absence of an estimated or real traffic matrix (due to infrastructure limitations mentioned in the previous section), we will use synthetic data expressing the aggregate bandwidth exchanged between discrete source-destination pairs. Based on this hypothesis, we will adopt the Profile-Based Routing (PBR) scheme. As it has been explained in section 2.1.1 PBR is built on the concept of traffic profiles. It should be clarified that our aim is not to evaluate the specified routing algorithm but to reveal through a set of experiments conducted how the network performance can be improved in terms of throughput and blocking probability even without having an accurate TM. Also, the simulation results reveal issues that should be seriously taken into consideration when traffic engineering is combined with traffic matrices. The PBR algorithm is compared with the conventional Shortest Path (SP) algorithm of IP networks, and the well-known Widest-Shortest Path (WSP) algorithm [17]. WSP selects the shortest path and if there are several such paths, it chooses the one with the largest residual bandwidth. This attribute makes WSP an interesting alternative to shortest-path routing and contributes significantly to load balancing especially when network infrastructure offers multiple shortest paths.

The simulation tests were carried out using the Network Simulator ns-2 [18]. SP and WSP implementations were already available in the platform. We implemented PBR on top of the existing QoS routing framework that supports explicit-route set-up which is an important prerequisite for the connection-oriented nature of PBR. Figure 3 shows a typical ISP topology that was used in our study. All links are bidirectional and have the same capacity with 1500 units of bandwidth. Link costs are set equal to 1. We consider the existence of 4 traffic profiles identified by the source-destination pairs (S1, D1), (S2, D2), (S3, D3), and (S4, D4) as depicted in Figure 4. For the PBR case, in order to make the offline pre-allocation of link capacities to the traffic profiles, we formulated the multicommodity network flow problem in AMPL modelling language [19]. To solve the problem we used the freely available NEOS optimization server [20]. In the online phase of the algorithm, we generated a random sequence of requests between the available traffic profiles. We assume that after connection establishment the amount of bandwidth allocated for a request is never freed again.

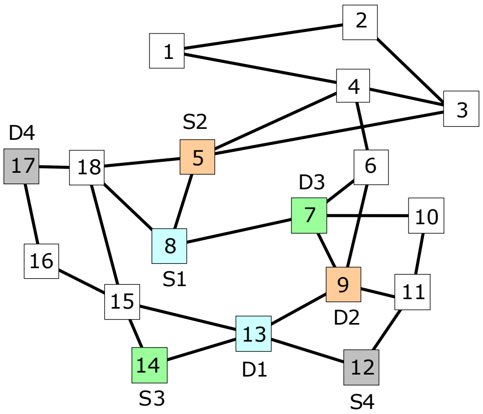

Figure 3. Simulation topology

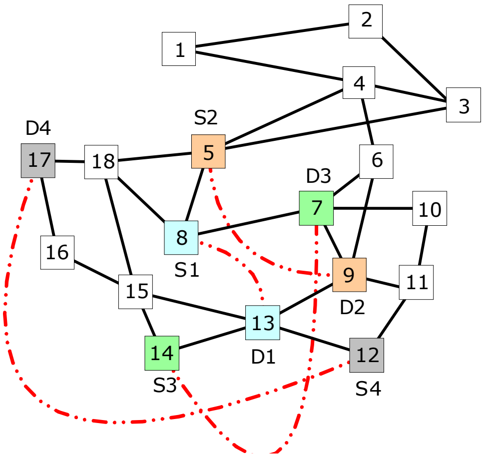

Figure 4. Simulation topology with excess edges

We examined two different simulation scenarios. In the first one, we solve the offline problem assuming infinite supply for each source-destination pair. Note that each profile is treated as a separate commodity. The objective is to maximize the total flow under the additional constraint (except for the flow conservation constraints) that all profiles have equal aggregate flow values. The solution obtained is that the maximum bandwidth (i.e. traffic profile) that can be routed for each ingress-egress pair is 2000 units. During the online phase the network is loaded with 3000 requests that they have bandwidth requirement uniformly distributed between 1 and 10 units (only integer values are selected). Also the requests are uniformly distributed among the 4 source-destination pairs. We use the total bandwidth of accepted requests and the number of blocked requests as the performance metrics to evaluate the behaviour of PBR in relation to WSP and SP algorithms. The results are presented in Figure 5 and they are the average of multiple test runs.

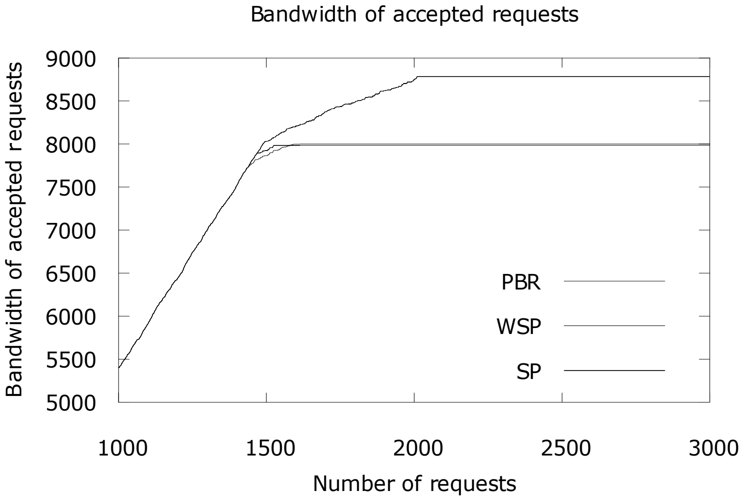

Figure 5. Total bandwidth of accepted requests as a function of the number of requests

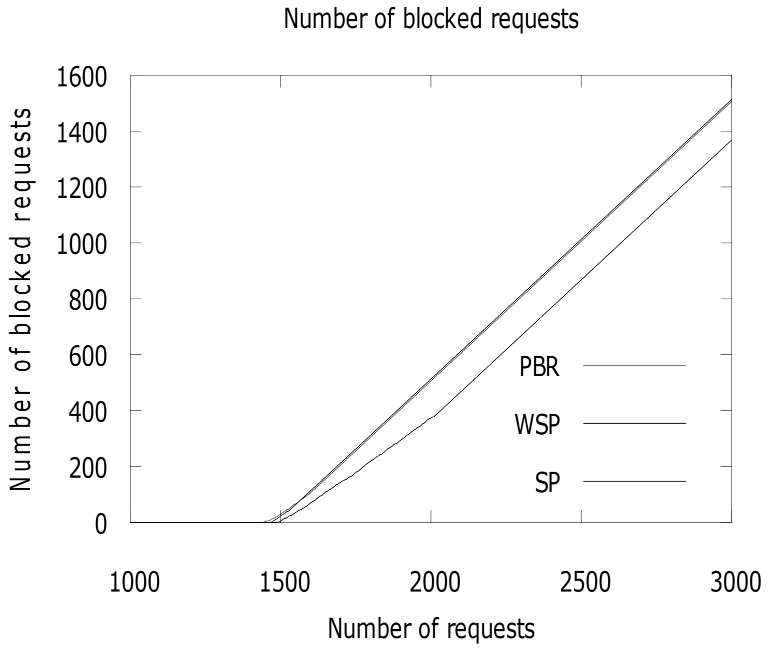

Figure 6. Number of blocked requests as a function of the number of requests

We see that WSP exhibits the best performance, reaching saturation point at 8700 units while in the cases of PBR and SP the network saturates at 8000 units. PBR shows poor performance but this is not surprising. First, although in the offline pre-processing the rough approximation of the aggregate bandwidth to be allocated was 2000 units for every source-destination pair, the system was loaded with much more traffic. The relatively low throughput is attributed to the fact that the source-destination pairs are deliberately selected to have high interference degree and that in the offline optimization we forced the total flow values of the profiles to be identical. As expected, in the SP routing case links on frequently used shortest paths are overutilized and they become increasingly congested.

Since the main goal of our simulation experiments is to study the effect of a priori knowledge of anticipated traffic, in the second simulation scenario we considered different traffic profile values. More specifically, the objective set in this test is to find paths to satisfy the maximum possible of 3000 units for each traffic class. In order to achieve a feasible solution for this problem instance, we added excess edges to the initial topology as illustrated in Figure 6. Each excess edge links the source and destination of a traffic profile with infinite capacity and cost. By this, we force traffic to go through the original network. The linear programming formulation gives 1500 units of bandwidth for profile 1 and 3 while 3000 units of bandwidth can be satisfied for profiles 2 and 4. In the online phase we load the network with 3000 requests with bandwidth requirement uniformly distributed between 1 and 10 units, similarly to the first scenario. However, the requests among traffic profiles are uniformly distributed in such a way that the total number of requests for profiles 2, 4 is double the number of those for profiles 1 and 3. In Figure 7 and Figure 8 we plotted the total bandwidth of accepted requests and the number of blocked requests as a function of the number of requests.

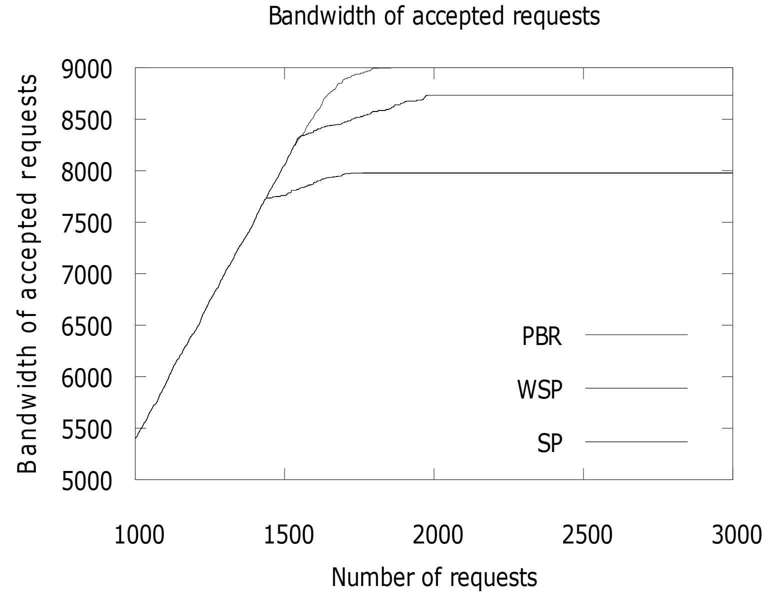

Figure 7. Total bandwidth of accepted requests as function of the number of requests

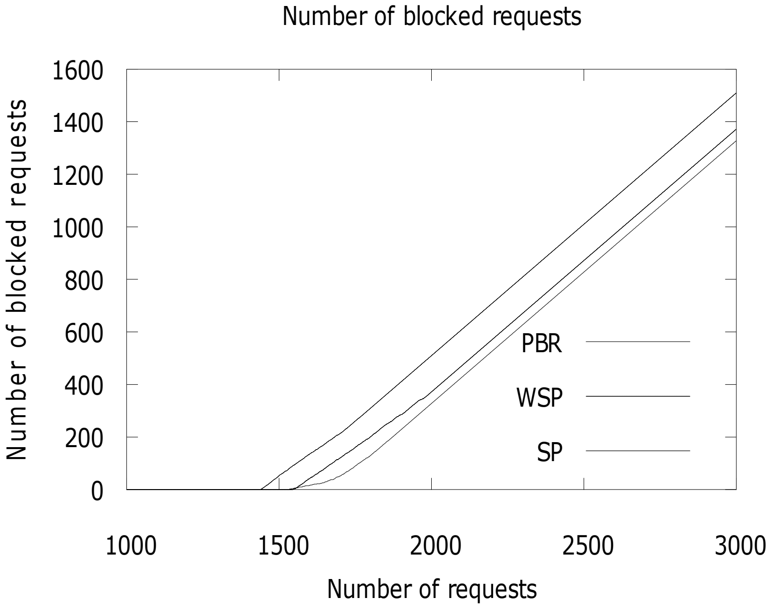

Figure 8. Number of blocked requests as a function of the number of requests

As depicted, PBR outperforms WSP and SP in terms of total bandwidth accepted and the number of rejected requests. Actually, PBR reaches the total throughput of 9000 units while WSP and SP saturate at 8733 and 7978 units respectively. In addition, PBR starts to block traffic after 1300 requests while WSP shows blocking effects after 1350 requests. SP algorithm is the first one to reject incoming traffic.

It is evident that the examined scenarios represent two different sides: the worst case and the best case scenario. The results acquired demonstrate that even in the worst case scenario a very rough estimation of the traffic profile (scenario 1) leads to performance close and not worst than that of SP. On the other hand, a very good approximation of the expected traffic (scenario 2) can significantly improve the network performance regarding load balancing, throughput and blocking probability. Also, we observe that aggregate values of bandwidth requested between ingress-egress pairs are not enough to determine the optimal offline allocation of link capacities to traffic profiles. Individual flow properties such as bandwidth values range are very crucial to the effective engineering of traffic.

5 Conclusions and remarks

In this paper we discussed the two fundamental categories of traffic engineering and how the knowledge of past monitoring information can become valuable input for accommodating future traffic requests. Besides, the notion of offline pre-processing expresses the current trend in the area of TE. As already mentioned, traffic statistics present a periodicity that allows the enforcement of a history-aware mechanism. A major issue in feeding routing decision schemes with past information is the determination of the parameters to be collected:

- First, the length of the measurement period that will supply the historical traffic statistics must be decided. Whether the monitored information will be gathered in long-term or short-term time scales is explicitly related to the function of TE where these traffic traces will be deployed.
- Second, we need to specify the polling frequency at which the defined measurements will take place. Despite the fact that traffic demands cannot be forecasted, Internet traffic falls into quite strong periodicities depending on the period of day. Hence, traffic statistics should be measured at such a frequency so as to capture data periodicities. A key factor in deciding the appropriate frequency is the study on the activities responsible for traffic generation: professional Internet use during day and entertainment scope of Internet use during evening.
- Another issue to be taken into consideration is the parameters which reflect the past traffic behaviour. It is apparent that the only pragmatic information that can be extracted through measurements is the bandwidth. Actually, it is the focal metric that drives route establishment procedure. In this area, a trade-off regarding the accuracy achieved in data attained and the ramifications on network operations should be weighted.

Coming now to the incorporation of the history aware mechanism into the TE framework, we should pay special attention to the seamless integration of such a scheme, in a way that it doesn't in any case overthrow network's stability. In this context, it should be offered as an add-on functionality to the existent QoS mechanisms, while, it's activation should be provided as an optional tool to the network administrator.

Last but not least, it has to be ensured that when deploying such a QoS mechanism, the usually gained performance will not deteriorate. For this, it is essential to deploy a flexible mechanism combining offline processing with an online phase in order to cope with unpredicted deviations from the precomputed scenarios.

Closing the paper, we should note the existence of several tools and methods for extracting raw measurements that can provide the historical information parameters required. Such tools vary from research oriented methods [12][13][14] to professional tools [15].

References

[1] W. Ben-Ameur. Multi-hour design of survivable classical IP protocols. *International Journal of Communication Systems*, 15:553-572, 2002.

[2] D. Awduche et al. Overview and principles of Internet Traffic Engineering. *RFC 3272*, May 2002.

[3] T. M. Thomas II. OSPF Network Design solutions. Cisco Press, 1998.

[4] J. Moy. OSPF Version 2. *RFC 2328*, July 1997.

[5] R. Callon. Use of OSI IS-IS for Routing in TCP/IP and Dual Environments. *RFC 1195*, December 1990.

[6] S. Suri et al. Profile-Based Routing: A New Framework for MPLS Traffic Engineering, *Computer Communications*, 24(4):351-365, March 2003.

[7] B. Fortz and M. Thorup. Internet Traffic Engineering by Optimizing OSPF weights. *In Proceedings of IEEE INFOCOM*, pages 519-528, March 2000.

[8] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS Weights in a changing world. *IEEE JSAC*, 20(4):756-767, May 2002.

[9] M. Ericsson, M. G. C. Resende and P.M. Pardalos. A genetic algorithm for the weight setting problem in OSPF routing. *Journal of Combinatorial Optimization*, 6:299-333, 2002.

[10] T. Ye et al. Dynamic Optimization of OSPF Weights using Online Simulation. *In Proceedings of IEEE INFOCOM*, June 2002.

[11] D. Awduche and B. Jabbari. Internet traffic engineering using Multi-Protocol Label Switching (MPLS). *Computer Networks*, 40(1): 111-129, September 2002.

[12] A. Medina et al. A Taxonomy of IP Traffic Matrices. *In Proceedings of SPIE ITCOM*, Boston, August 2002.

[13] M. Grossglauser and J. Rexford. Passive Traffic Measurement for IP Operations. *The Internet as a Large-Scale Complex System*, Oxford University Press, 2002.

[14] A. Medina et al. Traffic Matrix Estimation: Existing Techniques and New Directions. *In Proceedings of ACM SIGCOMM*, August 2002.

[15] Cisco Netflow. http://www.cisco.com/warp/public/732/Tech/nmp/netflow.

[16] TCP dump file analysis tool. http://jarok.cs.ohiou.edu/software/tcptrace/tcptrace.html

[17] R. Guerin, A. Orda, and D. Williams. QoS Routing Mechanisms and OSPF Extensions. *In Proceedings of Globecom*, Phoenix, AZ, November 1997.

[18] S. McCanne and S. Floyd. The LBNL Network Simulator, ns-2, http://www.isi.edu/nsnam/ns/.

[19] Robert Fourer and David M. Gay. Expressing Special Structures in an Algebraic Modeling Language for Mathematical Programming. *ORSA Journal on Computing,* 7: 166-190, 1995.

[20] NEOS optimisation server. http://www.neos.mcs.anl.gov/neos/.

[21] Jim W. Roberts. Traffic Theory and the Internet. *IEEE Communications Magazine*, 39(1): 94-99, January 2001.