King's Research Portal

DOI:
10.1016/j.comnet.2016.02.005

Document Version
Peer reviewed version

Link to publication record in King's Research Portal

Citation for published version (APA):
Jaron, A., Mihailovic, A., & Aghvami, A-H. (2016). QoS-Aware Multi-Plane Routing Method For OSPF-based IP Access Networks. *COMPUTER NETWORKS*, *99*, 1-14. https://doi.org/10.1016/j.comnet.2016.02.005

Citing this paper
Please note that where the full-text provided on King's Research Portal is the Author Accepted Manuscript or Post-Print version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version for pagination, volume/issue, and date of publication details. And where the final published version is provided on the Research Portal, if citing you are again advised to check the publisher's website for any subsequent corrections.

General rights
Copyright and moral rights for the publications made accessible in the Research Portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognize and abide by the legal requirements associated with these rights.

•Users may download and print one copy of any publication from the Research Portal for the purpose of private study or research.
•You may not further distribute the material or use it for any profit-making activity or commercial gain
•You may freely distribute the URL identifying the publication in the Research Portal

Take down policy
If you believe that this document breaches copyright please contact librarypure@kcl.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 20. Sep. 2020

QoS-Aware Multi-Plane Routing Method For OSPF-based IP Access Networks

Alexandre Jaron☆, Andrej Mihailovic,
and A.H. Aghvami, *Fellow, IEEE*

King's College London

Centre for Telecommunications Research

Strand, London, WC2R 2LS

Abstract

This paper presents a study of a method termed, Multi-Plane routing, that maximizes path diversity in IP routing and is targeted for IP access networks (AN). The motivation for the work is in the specific shortcomings of the conventional intra-domain IP routing principles such as ”shortest-path” and ”best-effort” when applied in IP ANs. We generalize these networks as the transit between the access routers and gateway and they range from a simple tree to meshed tree topologies. The method uses Multi-Topology OSPF standardized by the IETF and instantiates multiple OSPF installations in networks, each installation utilizing a portion of the topology in the conventional manner, i.e. routing plane (RP). Hence, all links are utilized by having at least one standard OSPF routing installation including them in the paths between access router and gateway. The method functions on extensions in routers and simple packet tagging allowing the routers to install and separate between paths of each RP. Routing is facilitated by the proposed method's algorithms for network planning and traffic engineering. The former is called the offline algorithm rendering the optimum number of RPs in an arbitrary topology by independently setting link weights for each plane. The latter is called the online algorithm that applies a policy-based routing scheme for dynamically selecting the best RP based on the introduced QoS-aware cost function. The paper concludes by significant improvements in throughput, packet loss rate, session

☆Corresponding author email address: alexandre.jaron@kcl.ac.uk

blocking and delays for numerous cases of topologies differing in numbers of networks nodes and degrees of meshing.

Keywords: Multi-Plane Routing, Traffic engineering, Open Shortest Path First, access networks, routing

1. Introduction

During the last decades the Internet has grown tremendously and has penetrated all aspects of everyday life. Although the Internet is based on a best effort service model, the simplicity of its packet-switched design and the flexibility of its underlying packet forwarding regime (IP) accommodate millions of users while offering acceptable performance. At the same time, exciting new applications and networked services have emerged, putting greater demands on the network. In order to offer a better-than-best-effort Internet, new service models that offer applications performance guarantees have been proposed. Many Quality of Service (QoS) aware networks are operating but there is still a lack of ubiquitous comprehension about the precise requirements of the applications such as voice-based services, 3DTV, real-time audio/video streaming, interactive video and gaming amidst others with varying Service-Level Requirements (SLRs). Moreover, the ever-increasing use of mobile devices places greater requirements on functioning of networks, especially access networks which connect residential, campus and small-business user to the core networks and the Internet. These networks have to scale up in bandwidth capacity to enable end-to-end service guarantees.

With the fast adoption of IP-based communications for mobile computing, users are expecting a similar service in wireless and wired networks. This raises the need for setting guarantees to the QoS offered service regardless of the access network technology or the mobility of terminals. The telecom world is moving towards an all-IP network, as IP is the dominant internetworking protocol in operation today. It becomes more and more recognized that using IP as the underlying infrastructure for next generation access networks makes strong economic sense and technical sense, both in installation and in operation, since it takes advantage of the ubiquitous installed IP infrastructure [1].

In light of these new expectations, research has raised questions on multipath diversity [2] in IP networks and naturally reassesses the shortest-path routing paradigm for the needs of the future networks. Perhaps these very needs have caused a discrepancy in deployment of all-IP networking including IP routing protocols all the way down to the edges of networks, that is, wireless access points in access networks. And from an IP development and deployment perspective, definition of access networks is rather unfounded. While cellular networks deliver IP services, telecom access networks run additional network layer routing solutions for fulfilling the needs of service deliveries while prudently nudging IP integration in their evolution. On the other hand, IP development has envisaged IP access networks for wireless terminals founded on IP routing in the network layer [3] and providing seamless mobility to the terminals [4]. Whether physically [3] or logically [5], IP access networks can be generally defined as the IP routing transit space in an administratively scoped network environment, bounded by the edges: gateway, providing connection to the Internet, and, access router, providing access to terminals. It is easy to imagine the opportunities for flexible deployment of IP access networks via rollouts and networking of wireless access points with technologies such as WiFi, femtocells and macrocells solutions.

Today, most access networks opt to deploy Cisco's Multi-Protocol Label Switching (MPLS) [6], which enables enterprises and service providers to build networks that deliver services over a single infrastructure. MPLS is a flow-based (also called connection-oriented) packet routing mechanism that assigns streams of packets to Label Switched Paths (LSPs). The most distinctive advantage of MPLS resides in its capability of arbitrary routing and splitting traffic. And it is this advantage of MPLS that makes it a more convincing solution for requirements posed in access networks, something that IP routing protocols fall short off. Yet, although effectively running as a supplementary routing solution in IP packet forwarding, MPLS often relies on IP routing protocols (such as Open Shortest Path First OSPF, intra-domain routing protocol) for computing LSP paths in networks. We also note some shortcomings of MPLS, mainly, its scalability and robustness issues as flows are mapped to dedicated LSPs. The overhead of building LSPs can be very high in relatively large-size networks due to large size of routing table and state information; MPLS introduces extra complexity of calculating, setting up and maintaining LSPs between every source-destination pair.

Our approach is based on OSPF routing protocol as the most widely used intra-domain routing protocol nowadays in backbone networks, large enterprise and data centers. OSPF is directly operating over IP and is an *adaptive* link-state protocol, i.e. each router within the network has a complete view of the network state and topology. Furthermore, OSPF is robust against element failures (e.g. node or link), flexible and scalable. However, OSPF does not allow arbitrary traffic splitting nor efficient path diversity as path alterations can be timely requiring changing of link weights and retransmitting the changes across the network.

Equal Cost MultiPath (ECMP) is a feature of OSPF, which many researchers investigated for path diversity that can achieve load balancing that is comparable to MPLS, by tuning link weights [7]. But in reality, ECMP only allows even splitting of traffic, which is not enough to provide a near-optimal and manageable performance comparable to that of MPLS or applicable to IP access networks. In the literature, many have conducted research to avoid problems associated with extra complexity of MPLS, link weight changes that trigger flooding of link-state messages, and even traffic splitting. Authors in [8], [9] proposed a new method based on Multi-Topology OSPF (MT-OSPF) [10]. Also, Wang. et al. [11] claimed that by partitioning the overall network demand into multiple subsets at the edge of the network so that each of them is delivered through dedicated IP routing planes, near-optimal performance could be achieved. However, previous conducted research considered multi-topology routing for transit and core networks. Furthermore, research conducted in [12] and [13], among many, used MT-OSPF for computing back-up routing topologies in case of failures, thus sub-topologies were not used simultaneously for forwarding traffic. The challenges and issues for IP access networks are not alike. Requirement for path diversity and dynamic traffic splitting are exalted due to many routing paths available for unidirectional packet flows between the gateway and the access routers. We consider the transit space to have arbitrary number of meshed routers as forwarding nodes, therefore this strictly follows the rule smart edge, simple core, rule that was originally designed for the Internet. Secondly, access networks, encounter high traffic variations due to the mobility of users, and variety of applications. Dynamic traffic engineering is hence required that can explore flexibility of path diversity and accommodate maximum levels of QoS for traffic flows.

In this paper, we extended and based our work on the research conducted by authors in [14] and [15], in which a link weight assignment algorithm for network planning and a traffic splitting adjustment algorithm have been developed for creating up to five OSPF routing planes on one hand, and then spreading traffic amid them following the rule same path for same flow. Routing plane is an installation of the standard OSPF routing protocol. In IP access networks with various degrees of topology meshing, optimum number of OSPF routing planes should utilize all links in the network for path diversity of traffic. Hence, the solution is based on OSPF with no major changes to the operation of the protocol, only extensions to support multiple planes in the networks. The solution also relies on network planning and traffic engineering of multiple planes. To our knowledge, quality of service has never been considered using the Multi-Plane Routing (MPR) approach. Also, no routing policy on routing plane selection for a new incoming session, based on real traffic data, has been proposed.

1.1. Contributions

Towards this end, the contribution of this paper is threefold. First, we imagined an offline algorithm for creating an optimal set of routing planes that is topology independent. The offline algorithm presents a network planning tool building the planes based on independent distribution of link weights for each plane. This offline algorithm has for sole input the physical topology with the associated link capacities. It is an extension to the algorithm presented in [14] and has been performed under Matlab. Second, we developed a QoS-aware cost function for routing plane state monitoring that we implemented and extended to network simulator NS-2, as well as developing a whole package (enabling MT-OSPF) on top of the basic Link-State module present in NS-2. Third, we created a policy-based routing scheme for access networks, as a traffic engineering tool, that selects the best routing plane for providing QoS to a new user while improving network performance.

1.2. Organization

The rest of the paper is organized as follows. In Section 2 the MP routing approach is presented where the offline network planning algorithm used to find an optimal set of routing planes is shown. The formal model of our proposed QoS-aware MPR (Q-MPR) mechanism, processed online, is described in Section 3. The performance evaluation of both offline and online algorithm along with the implementation issues are depicted in Section 4. Finally, Section 5 concludes the paper.

2. Theoretical Foundation: MPR

In this section, we introduce the multi-plane routing strategy. We start by describing its principle and secondly, we present the algorithm for building the routing planes with the objective of maximizing path diversity.

2.1. MPR method: Overview

MPR allows the routers within an area to maintain several *independent* logical planes, with independent set of link weights, and hence independent routing tables for each routing plane (RP). Each RP is an instance of OSPF from which a subset of the physical links have been removed for carrying traffic. Therefore, an RP is a subset of the underlying network (or physical topology). It can overlap with another or share any subset of the underlying network. In standard OSPF, as shown on the left-hand side of Figure 1, one routing information base (RIB, or routing table) is extracted from the topology database, and subsequently, one forwarding information base (FIB, or forwarding table) is used. With MPR, bold lines in Figure 1, it is not one RIB and one FIB that are used but instead, one RIB/FIB per plane. Data traffic is mapped to a specific routing plane that a router selects, and is routed according to the corresponding RIB. It is outside the scope of this paper to specify how the information in various plane specific forwarding structures is used during packet forwarding or how incoming packets are associated with the corresponding routing plane.

Figure 2 depicts a simple example of how four routing planes can be set up in a simplistic topology. The left subfigure shows the path between source S and destination T in all four routing planes whereas the right subfigure indicates a possible link weight configuration for one of the routing plane.

The cost of a path, which is the sum of the link weights along the path, has to be the lowest for this path in order to be considered a shortest path (OSPF).

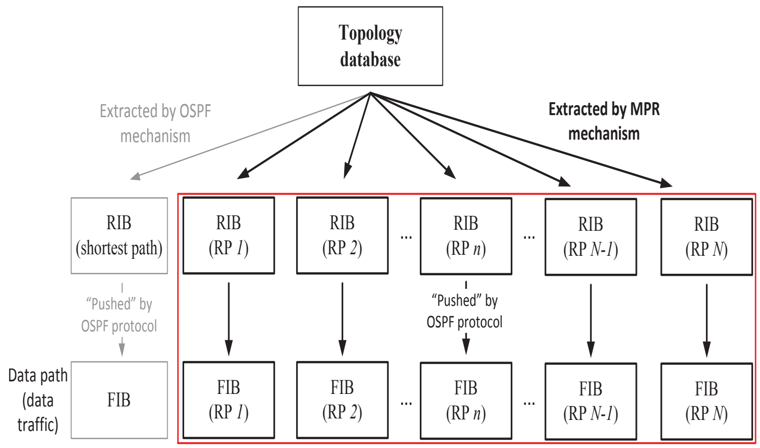


Figure 1: Data flow under conventional routing (left-hand side, thin grey lines) and under MP routing (thick black lines).

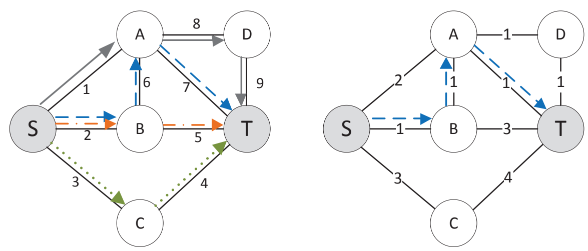


Figure 2: A simple example of 4 RPs. Numbers indicate link IDs (left) and link weights for one RP (right).

2.2. *Graph-theoretical MPR principle*

For a given communication access network, consider its topology to be mapped to the corresponding directed connected graph $G = (\mathscr{V}, \mathscr{E})$. The network consists of a set $\mathscr{E}$ of E ($\mathscr{E} : e = 1, ..., E$) bidirectional edges with finite capacities $C = (C_e,\ e = 1, ..., E)$ and a set $\mathscr{V}$ of V ($\mathscr{V} : v = 1, ..., V$) vertices. Let $\mathscr{N} : n = 1, ..., N$ be the set of routing planes and each edge $e \in \mathscr{E}$ be assigned with $|N|$ distinct link weights (denoted by $w_n(e),\ n \in \mathscr{N}$) . The network also supports a set $\mathscr{D}$ of D ($\mathscr{D} : d = 1, ..., D$) demands or gateway (GW)-access router (AR) pairs. For example, $d = 1$ is the pair *GW-AR1*. Let also $\mathscr{P}$ be the total set of available paths for each pair d in all RPs in $\mathscr{N}$ (we consider only symmetric routing). And let $P_n^d \in \mathscr{P}$ be the set of acyclic paths for demand d and routing plane n according to the link weight configuration W_n

for that routing plane. They are represented by an $N \times E$ matrix R^d , where $R^d_{en} = 1$ if path of pair d uses link e in routing plane n, and $R^d_{en} = 0$ otherwise. The overall routing matrix, whose dimension is $E \times (N \times D)$, is given by:

$$R = \begin{bmatrix} R^1 & R^2 & \cdots & R^D \end{bmatrix} \tag{1}$$

Consider Figure 2 as an example. The corresponding routing matrix is:

$$R^1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T \tag{2}$$

The general routing matrix for demand d can now be formerly rewritten as follows:

$$R^d = \begin{bmatrix} R^d_{11} & R^d_{21} & \cdots & R^d_{E1} \\ R^d_{12} & R^d_{22} & \cdots & R^d_{E2} \\ \vdots & \vdots & \ddots & \vdots \\ R^d_{1N} & R^d_{2N} & \cdots & R^d_{EN} \end{bmatrix}^T \tag{3}$$

With $\forall d \in \mathscr{D},\ \forall n \in \mathscr{N},\ and \forall e \in \mathscr{E}$.

The link weight assignment mechanism is based on [14], which aimed at maximizing path diversity in the network. The mechanism uses a cost function denoted *Path Diversity Index (PDI)* that defines, for each demand $d \in \mathscr{D}$ and for each link $e \in \mathscr{E}$, the number of planes that include e in their shortest paths for demand d (between each GW-AR pair). It was expressed as the following:

$$\textbf{\textit{PDI}}^d_e = \sum_{n \in \mathscr{N}} R^d_{en},\ \forall e \in \mathscr{E} \tag{4}$$

The ultimate objective is to minimize the chance that for a given demand all routing planes share a single link; secondly, to maximize the chance that any single link is used in at least one plane. The reason for this if congestion or failure occurs the associated demand can avoid this critical link and also, to ensure the link will not be left unutilized for carrying traffic. The algorithm in [14] then introduces *Full Path Diversity Index*

(FDPI) which designates whether a critical link e is included in the shortest paths for demand d in all routing planes. *FPDI* is equal to 1 if $PDI_e^d = |N-1|$ (link e is not present in at least one RP, refer to Subsection 2.3) and 0 otherwise. In summary, the link weight assignment problem is formally described as follows: to calculate $|N|$ sets of positive link weights $W_n = \{w_n(e)\} : 1 \leq w_n(e) \leq K$, with $\forall n \in \mathscr{N}$, $\forall e \in \mathscr{E}$ and K $(= 2^{16} - 1)$ the highest weight value that OSPF can handle, in order to maximize:

$$\sum_{d \in \mathscr{D}} \sum_{e \in \mathscr{E}} \mathit{FPDI}_e^d \tag{5}$$

The link weight assignment problem finally returns, for routing plane n, the link weight configuration W_n $(1 \times E)$ as follows:

$$W_n = \left[\begin{array}{cccc} w_n(1) & w_n(2) & \cdots & w_n(E) \end{array} \right] \tag{6}$$

Hence, the overall link weight setting for graph G with N routing planes can be expressed as follows:

$$W = \left[\begin{array}{cccc} W_1 & W_2 & \cdots & W_N \end{array} \right] \tag{7}$$

2.3. Topology independent RP construction

This section describes the offline algorithm for the routing plane construction. As stated previously, the ultimate objective is to maximize the diversity in terms of available paths for each GW-AR pair between all routing planes. In order for the algorithm to be effective whatever the input, namely physical topology, we used two baseline tree-shaped topologies, in which the meshing degree took different values, that is the node degree distribution. Indeed, the average node degree will have a direct impact on the algorithm performance as the higher the node degree distribution, the more available paths for each GW-AR pair, hence the more routing planes can be found. The algorithm starts by computing the first plane using the InvCap method proposed by Cisco. Invcap sets the link weights to the inverse of the capacity of the links. Simply, for each link $e \in \mathscr{E}$, $w(1, e) = 1/C_e$. Figure 3 shows one of the topologies used for the simulations. Please note that the depicted topology is one of the baseline topologies, to which a different meshing degree is applied to create several sub-topologies.

Link capacities are set up depending on the level they belong to. For instance, links connecting the gateway with next-hop nodes belong to one level, which we will call Level 1. Thus, link capacities are randomly generated following a uniform distribution in $[360, 400]$ for Level 1, $[200, 240]$ for Level 2, $[140, 180]$ for Level 3 and $[60, 100]$ for Level 4 in topology 1. Topology 2 comprises five levels, therefore, link capacities are generated in the following intervals: $[360, 400]$ for Level 1, $[160, 200]$ for Level 2, $[110, 150]$ for Level 2 and 3, and finally $[50, 90]$ for Level 5.

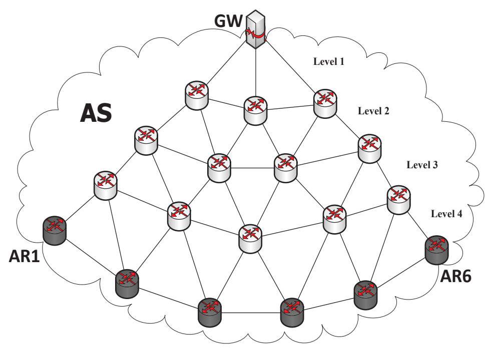


Figure 3: Instance of one of the sub-topologies for Topology 1 used in simulations.

In our approach, we distinguish between the default plane which is the standard flat OSPF network topology where all the links can be used for carrying traffic and the routing planes where a set of links are excluded from the routing process. Three rules are used in the algorithm, they are listed below:

1. Each link must not be used for routing in at least one routing plane.
2. All planes are connected which means, in each plane, there is a valid route for each gateway (GW)-Access Router (AR) pair. All nodes in between are considered transit routers, they are not traffic sources or sinks.
3. Each link is used in at least one plane. This property ensures maximum path diversity.

Figure 4 sums up the offline process for finding and constructing the optimal set of routing planes. In order to create the optimal set of routing planes for each topology, three methods are used. As mentioned above, a first plane is created, RP 1, whose

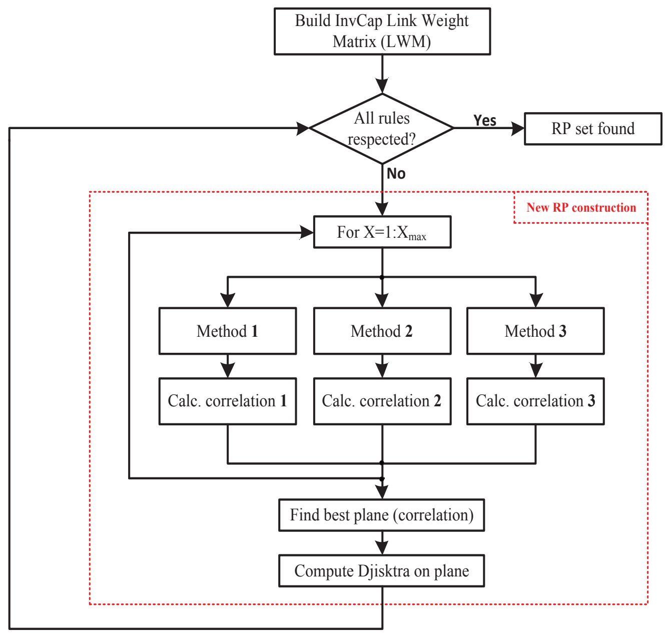


Figure 4: State diagram of the offline RP construction algorithm.

link weight setting is calculated using the inverse of the capacity of all the links in the network. Obviously, one plane is not enough to satisfy all three rules, so a new plane needs to be found. The design of a new plane is based on finding a link weight configuration. Three methods for computing the link weights are used.

Method 1. Iterative plane construction

The method determines the cost of each link to create a new plane. The cost takes into account the inverse of the link capacity, the averaged link cost of the $N-1$ planes, and a third argument.

$$w_n(e) = \frac{\max\limits_{e \in \mathscr{E}} (C_e)}{C_e} + \frac{1}{N} \sum_{n=1}^{N-1} w_n(e) + \alpha_e(n).X \tag{8}$$

With $\forall e \in \mathscr{E}$, $\forall n \in [1, N-1]$. X is a multiplicative parameter that is used to vary the granularity of the method; that is, the higher the value of X, the more routing planes will be tested. X ranges from 1 to X_{max} by step of 1, with $X_{max} =$

$\{2;\ 4;\ 8;\ 16;\ 32;\ 64\}$. α_e is defined as follows:

$$\alpha_e(n) = \begin{cases} 1, & \text{if link } e \text{ is included in a path in RP } n-1; \\ 0, & \text{otherwise.} \end{cases}$$

Method 2. Link degree of involvement

Unlike Method 1 that only considers the involvement of a link in RP $N-1$, Method 2 considers the involvement of a link e in all RP $n \in [1, N-1]$. The link cost for Method 2 is defined as follows:

$$w_n(e) = \frac{\max\limits_{e \in \mathscr{E}} (C_e)}{C_e} + \frac{1}{N} \sum_{n=1}^{N-1} w_n(e) + \beta_e(n).X \qquad (9)$$

With $\forall e \in \mathscr{E}, \forall n \in [1, N-1]$ and with $\beta_e(n) = \sum\limits_{n=1}^{N-1} \alpha_e(n)$.

Method 3. Max link degree involvement per GW-AR pair

Method 3 is basically a sub-set of Method 2, where the cost of a link e that is the most used in one RP is penalized.

$$w_n(e) = \frac{\max\limits_{e \in \mathscr{E}} (C_e)}{C_e} + \frac{1}{N} \sum_{n=1}^{N-1} w_n(e) + \gamma_e(n).X \qquad (10)$$

With $\forall e \in \mathscr{E}, \forall n \in [1, N-1]$. $\gamma_e(n)$ penalizes the cost of the link that is the most used in all routing planes $N-1$ for each GW-AR pair. And $\gamma_e(n) = \max\limits_{d \in \mathscr{D}} \left(\sum\limits_{n=1}^{N-1} \alpha_e^d(n) \right)$.

Note that the value of N changes every time a new routing plane has to be found in order to satisfy the three aforementioned properties. For instance, the value of N is equal to 1 when the algorithm starts and builds the first RP based on the InvCap method, then $N = 2,\ 3,\ ...$ until a minimum set of routing planes that satisfy all three rules is found. In order to select the best plane, in terms of maximum path diversity, among all tested routing planes, we use the Pearson product-moment correlation coefficient. After finding a new plane, the algorithm calculates the correlation of the new plane n and the previously constructed ones, $n-1$ planes, with the physical topology that we denote N_0. We chose not to calculate the correlation between RPs two by two as one RP can be uncorrelated with a second one, and a third RP can present a high correlation with the first one. All RPs are compared with the physical topology which never changes. Let $\hat{\mathscr{N}} \subseteq \mathscr{N}$ be a subset of the optimal set of routing planes. Therefore, the Pearson coefficient can be expressed as follows:

$$\begin{aligned} \zeta_{\hat{\mathscr{N}},N_0} = \text{corr}(\hat{\mathscr{N}}, N_0) &= \frac{\text{cov}(\hat{\mathscr{N}},N_0)}{\sigma_{\hat{\mathscr{N}}}\sigma_{N_0}} \\ &= \frac{E\big[\big(\hat{\mathscr{N}}-\mu_{\hat{\mathscr{N}}}\big)\big(N_0-\mu_{N_0}\big)\big]}{\sigma_{\hat{\mathscr{N}}}\sigma_{N_0}} \end{aligned} \tag{11}$$

After calculating the correlations for the three methods, we have therefore three contending routing planes. This process is then iterated in the loop taking values from 1 to X_{max} by step of 1. Once X_{max} is reached, the offline algorithm computes the minimum, mean and standard deviation of all calculated correlations for all three methods and select the plane with the lowest correlation. Djisktra's algorithm is then performed to compute the paths on the selected routing plane based on the link weight configuration taken from the method. The algorithm stops when a minimum set of routing planes satisfy all three properties.

3. QoS-aware MPR

In this section, we integrate QoS awareness to the MPR mechanism for traffic engineering. This section describes how routing planes are monitored and how the routing plane selection is performed.

3.1. Multi-Constrained Plane

The network is constructed to support a set $\mathscr{U}$ of U ($\mathscr{U} : u = 1, ..., U$) users. For simplicity, let N^u be the paths in all routing planes for user u (demand d). For every u, we define an $N^u \times 1$ vector $\pi^{u,d}$ with the rate $\pi_n^{u,d}$ of user u using RP n as the uth entry of $\pi^{u,d}$. The total rate of user u is denoted $\|\pi^u\|$. Let a $\sum_u N^u \times 1$ vector π^d represent the total bandwidth request at an access router for demand d:

$$\pi^d = \left[(\pi^1)^T \; (\pi^2)^T \; \cdots \; (\pi^U)^T\right]^T \tag{12}$$

Finally let π be the total aggregated traffic in the network and it is expressed as $\pi = \sum_d \sum_u \pi^{u,d}$. We consider the routing planes to be identical for downlink and uplink however they can be selected differently for downlink and uplink. Also, a stream of packets belonging to the same session will follow the same path (same RP) for session request and transfer of actual data.

Each access router has a utility function U^d as a function of its aggregate demand $\sum_u \|\pi^{u,d}\|$. The basic multi-plane routing problem is to maximize the network resources by allocating a specific routing plane, that is a specific path, for each user u of rate $\|\pi^u\|$ subject to link capacity constraints. Let $\|\pi^{u,d}\|_0$ be the number of non-zero entries of $\pi^{u,d}$. Then the multi-plane problem can be formulated as a non-convex optimization problem:

$$\begin{aligned} \max_{\pi \geq 0} \quad & \textstyle\sum_d U^d(\|\pi^{u,d}\|), \ \forall d \in \mathscr{D} \\ \text{s.t.} \quad & R\pi \leq c \\ & \|\pi^{u,d}\|_0 = 1, \ \forall u \in \mathscr{U}, \ \forall d \in \mathscr{D}. \end{aligned} \tag{13}$$

Bandwidth constraint Let a path p_n^d be represented as a concatenation of successive links, and $p_n^d = \{e_{ij,n} \mid \forall i \neq j, \ (i, \ j) \in \mathscr{V}^2, \ \forall d \in \mathscr{D}, \ \forall n \in \mathscr{N}\}$. We denote by $b(e_{ij,n})$ the available bandwidth on edge e_{ij} for demand d in RP n. Therefore the available bandwidth of the path p_n^d in RP n for demand d is:

$$b(p_n^d) = \min_{e_{ij,n} \in p_n^d} b(e_{ij,n}) \tag{14}$$

We note c_b the QoS bandwidth constraint for the session. Then the bandwidth requirement is expressed as:

$$b(p_n^d) \geq c_b \tag{15}$$

Bandwidth is a non-additive QoS parameter, therefore it is easily dealt with a pre-processing phase by pruning all paths that do not satisfy the QoS requirements for the session [16].

Additive constraints As discussed in prior sections, considering just one QoS constraint at a time is not sufficient to provide QoS guarantees to all kinds of applications, especially ever-increasing Internet multimedia applications. Thus we propose to use the principle of multi-constrained path or MCP QoS routing [17] based on multiple QoS metric to find a feasible path (routing plane) for each GW-AR pair. Each application has different service-level requirements, some are delay-, jitter- and/or reliability-sensitive applications, thus, this approach can provide more on-demand and dynamic support for all types of traffic.

Each link $e_{ij,n}$ in path p_n^d is associated with K additive QoS metrics $m_k(e_{ij,n})$, where $k \in \kappa$ $(\kappa : k = 1,\ 2, \ldots,\ K)$. There are also K constraints c_k^t, $\forall t \in \tau$, where τ $(\tau : t = 1,\ 2, \ldots,\ T)$ is the set of traffic types. The MCP problem is to find RP n for demand d, that is between access router r and the gateway, that satisfies the following requirement:

$$m_k(p_n^d) \equiv \sum_{e_{ij,n} \in p_n^d} m_k(e_{ij,n}) \leq c_k^t,\ \forall k \in \kappa \tag{16}$$

without cost optimization (primary cost of feasible path p_n^d in routing plane n satisfying requirement (16) is not necessary to be minimized).

The non-linear cost function [15], [17] shown in (17) illustrates the method to non-linearly combine additive QoS parameters, such as delay, jitter, reliability, packet loss, into a single cost metric for any path p_n^d in routing plane n for demand d while the non-additive ones such as bandwidth, as stated previously, is easily dealt with a pre-procession step. Let Γ $(\Gamma : \gamma = \gamma_0,\ \gamma_1, \ldots,\ \gamma_k)$ be the set of weights used for each constraint k. Therefore, the cost function for any path p_n^d for demand d in routing plane n is expressed as follows:

$$\begin{aligned} \varphi_\Gamma^t(p_n^d) &\equiv \left(\tfrac{m_1(p_n^d)}{c_1^t}\right)^{\gamma_1} + \left(\tfrac{m_2(p_n^d)}{c_2^t}\right)^{\gamma_2} + \ldots + \left(\tfrac{m_k(p_n^d)}{c_k^t}\right)^{\gamma_k} \\ &= \sum_{i=1}^{k} \left(\frac{m_i(p_n^d)}{c_i^t}\right)^{\gamma_i} \end{aligned} \tag{17}$$

With $\forall d \in \mathscr{D},\ \forall n \in \mathscr{N},\ \forall t \in \tau$ and $\gamma_i \in [0,\ 1]$.

As mentioned above, φ_Γ^t is a cost function weighted by the set Γ. The γ_i variables allow to give more priority to specific QoS parameters than others, for instance, certain multimedia applications require drastically low delay or jitter but may be more tolerant to packet loss.

3.2. Plane Selection Policy for Q-MPR

Though the proposed QoS-aware multi-plane routing scheme allows an incoming session to be routed along a certain routing plane that respects the service level agreements for the session, it does not yet guarantee that the load is optimally balanced within the network, and hence network is not well utilized. In order to ensure that low QoS traffic is routed through lesser congested paths away from the paths in routing planes used by greedy QoS sessions, we propose a plane selection (PS) policy. Policy PS has been enforced to ensure traffic within the network is regulated and routed appropriately [18]. The aforementioned routing policy needs to be implemented in the border routers within the network, namely the access routers and the gateway. This policy assures that a routing plane is selected by these border routers according to the class of traffic an incoming packet belongs to.

To derive the routing policy we define extra notations. Let χ be the subset of routing planes ($\chi \subseteq \mathscr{N}$) that support the quality of service required by the session. Therefore, we denote $\overline{\chi} \subseteq \mathscr{N}$ the complimentary set of χ which denotes the routing planes that do not provide QoS guarantees for a new incoming session. Note that $\chi \cup \overline{\chi} = \mathscr{N}$. In the case where several routing planes respect the SLRs for the session, one still has to be selected. Towards this end, we add an extra parameter in Equation (17) that checks the available bandwidth after considering the current throughput request of a new session. Thus, Equation (17) becomes:

$$\varphi^t_\Gamma(p^d_n) \equiv \sum_{i=1}^{k} \left(\frac{m_i(p^d_n)}{c^t_i} \right)^{\gamma_i} + \left(\frac{b(p^d_n) - \|\pi^{u,d}\|_0}{C(b(p^d_n))} \right)^{-1} \tag{18}$$

With $\forall d \in \mathscr{D}$, $\forall n \in \mathscr{N}$, $\forall t \in \tau$ and $\gamma_i \in [0,\ 1]$.

Note that $C(b(p^d_n))$ represents the capacity of the link $e_{ij,n} \in p^d_n$ that has the least available bandwidth on the path in routing plane n for demand d. Equation (18) allows the AR to select the least congested routing plane, i.e., the one that presents the highest available bandwidth after taking into account the required throughput of the new session request. The overall decision making process is depicted in algorithm 1 which presents the overall plane selection policy for the Q-MPR mechanism.

When a packet arrives at an AR r, the policy routing procedure is performed. If the session is admitted into the network, AR r verifies which traffic class the incoming session belongs to and obtains SLRs for that particular traffic class (shown in Table 3). The AR discards all RPs that do not satisfy the QoS constraints; at that point we know there is at least one RP that can be selected for carrying the session. Set χ is retrieved,

Algorithm 1 Plane Selection Policy

```
procedure POLICY-PS ($P_n^d, \tau, \kappa, \mathscr{N}, V$)
    Packet arrives at AR $r \in \mathscr{V}$ destined to gateway $g \in V$
    if $\|\pi^{u,d}\|_0 \leq c_b$, for at least one $n \in \mathscr{N}$ then
        Session is admitted
        Perform lookup and check traffic class $t \in \tau$
        Obtain QoS requirements $c_k^t$ for traffic class $t$
        Prune all RPs in $\mathscr{N}$ that do not satisfy SLRs for each $k \in \kappa$ and retrieve set $\chi$.
        Calculate cost for each RP $x_i \in \chi$
        $\varphi_\Gamma^t(p_{x_i}^d) = \sum_{i=1}^{k} \left(\frac{m_i(p_{x_i}^d)}{c_i^t}\right)^{\gamma_i} + \left(\frac{b(p_{x_i}^d) - \|\pi^{u,d}\|_0}{C(b(p_{x_i}^d))}\right)^{-1}$
        Select RP $x_1$ with lowest cost $\varphi_\Gamma^t$ for the session of user $u$ so that
        $\varphi_\Gamma^t(x_1) \leq \varphi_\Gamma^t(x_2) \leq \ldots \leq \varphi_\Gamma^t(N - \overline{X})$
    else Reject session
    end if
end procedure
```

and the cost is calculated for each RP in χ. The RP presenting the lowest cost φ_Γ^t is selected.

4. Performance Evaluation

How well can our new Q-MPR scheme perform, and how fast can the optimal set of routing planes (RPs) be? In this section, we compare the performance of the Q-MPR mechanism against currently deployed link-state routing protocols, OSPF, Cisco's InvCap and our basic MPR method, with no routing plane selection policy based on QoS.

4.1. Offline algorithm

This subsection details and evaluates the performance obtained by the offline procedure of constructing an optimal set of routing planes. The simulations are performed with Matlab for the offline algorithm and comprise eleven different topologies. We generate two main topologies in which we use a different meshing degree, spanning from a strict tree sub-topology to an almost full-meshed topology. Table 1(a) presents the setup of all the eleven topologies used for simulations. "$TxMy$" indicates the topology number and the degree of meshing. The higher y, the higher node degree distribution, and hence the more paths will be available between each GW-AR pair. $T1M1$ and $T2M1$ are sub-topologies of Topology 1 and Topology 2 where only one path is available for carrying traffic for each GW-AR pair.

Topo	# Nodes	# ARs	# Links	Total capacity (Gb)
T1M1	19	6	18	7.84
T1M2	19	6	32	11.94
T1M3	19	6	36	12.98
T1M4	19	6	39	14.06
T1M5	19	6	41	15.34
T2M1	32	14	31	9.84
T2M2	32	14	53	15.28
T2M3	32	14	59	16.48
T2M4	32	14	61	16.88
T2M5	32	14	65	18.00
T2M6	32	14	67	18.40

(a) Setup of the topologies

Topo \ X	2	4	8	16	32	64
T1M1	1	1	1	1	1	1
T1M2	-1	-1	-1	5	4	4
T1M3	-1	-1	-1	7	7	5
T1M4	-1	-1	-1	5	6	4
T1M5	-1	-1	-1	5	6	5
T2M1	1	1	1	1	1	1
T2M2	-1	9	6	4	3	3
T2M3	-1	-1	-1	5	5	4
T2M4	-1	-1	-1	6	7	4
T2M5	-1	-1	-1	8	6	5
T2M6	-1	-1	-1	-1	5	5

(b) Output of the offline algorithm

Table 1: Offline setup and performance.

Topo \ Tech	OSPF	InvCap	MPR					
			X=2	X=4	X=8	X=16	X=32	X=64
T1M1	0.0129	0.0179	0.158	0.224	0.169	0.138	0.137	0.143
T1M2	0.0175	0.016	0.259	0.352	0.540	0.598	0.755	1.080
T1M3	0.0121	0.0116	0.251	0.358	1.057	0.861	1.354	1.495
T1M4	0.0118	0.0172	0.252	0.455	0.518	0.707	1.194	1.142
T1M5	0.012	0.0126	0.560	0.373	0.518	0.641	1.120	1.511
T2M1	0.0753	0.0771	0.297	0.371	0.291	0.292	0.308	0.286
T2M2	0.0851	0.0745	1.081	1.680	1.359	1.203	1.612	2.131
T2M3	0.0785	0.0751	1.437	1.393	2.025	1.599	2.424	3.060
T2M4	0.0734	0.0756	1.074	1.280	1.926	2.540	3.439	2.975
T2M5	0.1032	0.076	1.093	1.286	1.908	3.082	2.114	3.897
T2M6	0.0779	0.076	1.253	1.285	1.792	3.009	2.387	4.781

Table 2: Offline algorithm complexity, running time (s)

Table 1(b) shows the output of the offline algorithm, that is the number of RPs found to form an optimal set on a per topology basis. Different values of X have been studied, and Table 1(b) clearly indicates that for a value of 64, generally an optimal set with fewer RPs is found. Recall that an optimal set of RPs is found if the three properties stated in Section 2.3 are satisfied with a minimum number of RPs. A value of -1 denotes an error, the algorithm stopped and no optimal set was found. In this case, the value of X is increased to the next value, and the process starts over. Note that 1 routing plane could be found for $T1M1$ and $T2M1$ as these two topologies are strict trees, therefore only one path is available between each access router and gateway pair. Among all values of X tested, the set with the fewest number of routing planes is selected, this is to ensure minimum implementation and routing table maintenance overhead. However, with a higher number of planes, more paths are available and thus one can assume that traffic can be better balanced. We will show in subsection 4.2 that this statement is wrong.

The computational complexity, represented by the running time expressed in seconds, is shown in Table 1(c). OSPF and InvCap methods outperform our proposed strategy as only one path is computed for each GW-AR pair. It can also clearly be seen that the higher the value of X, the longer and the more complex the algorithm is. Also, as the topology presents a higher meshing degree, namely more paths are available for each GW-AR pair, the complexity is increased. A maximum value of 64 is shown for X as the algorithm does not perform better for higher values of X.

4.2. Online algorithm

In this subsection, the performance of the online algorithm, which takes for input the optimal set of routing planes computed in the offline algorithm, is studied. The routing plane selection and thus the splitting of traffic is directly affected by the output of the RP construction process. The online simulations were run using the well known network simulator NS-2 that we extended to support Multi-Topology OSPF routing, as specified by the IETF [10]. The extensions to NS-2 to support multi-plane routing have been studied by authors in [19, 20]. Different classes of traffic have been used for simulations, each associated with specific QoS requirements or SLRs, and are all listed in Table 3.

The routing plane configuration drawn from the offline process, which determines the link weight matrix (LWM) for each RP, is computed and constructed in NS-2 . Recall that each routing plane is a subset of the physical topology and each is associated with a separate routing table. A new incoming session is generated randomly among traffic classes shown in Table 3. As the simulation runs, traffic is generated with a decreasing session arrival time so as to load the network until congestion level. When a new session request is made at an access router, the latter checks for bandwidth availability on the path(s) to reach the destination, independently of the method used (OSPF, InvCap, MPR or Q-MPR).

OSPF and InvCap protocols will forward the traffic demand to the destination on the available path. With MPR, several routing planes, hence several paths are available towards the destination node (GW for uplink, AR for downlink). For each new incoming session, a routing plane is randomly selected for routing traffic towards the destination. In Q-MPR, new sessions are forwarded based on the required QoS for the sessions.

Table 3: Traffic types[1] and associated QoS requirements.

Traffic Class	Data Rate	Mean Duration	QoS requirements		
			Latency	Jitter	Packet loss
Class 1	Low (≈ 150 Kbps)	180 sec	40-65 ms	0.5-2 ms	0.1-0.5 %
Class 2	Medium (≈ 250 Kbps)	300 sec	4-5 s	*none*	5 %
Class 3	Low (≈ 128 Kbps)	200 sec	300-600 ms	2 ms	5 %
Class 4	High (≈ 500 Kbps)	360 sec	300 ms	30 ms	1 %
Class 5	Low (≈ 100 Kbps)	90 sec	*no specific requirement*		

[1] Applications examples; Class 1 : VoIP, Class 2 : streaming video, Class 3 : streaming audio, Class 4 : interactive video, Class 5 : best effort data.

Planes not satisfying all QoS requirements will be pruned at session arrival. In the case where several RPs satisfy the QoS requirements for the session, the plane with most available bandwidth will be utilized.

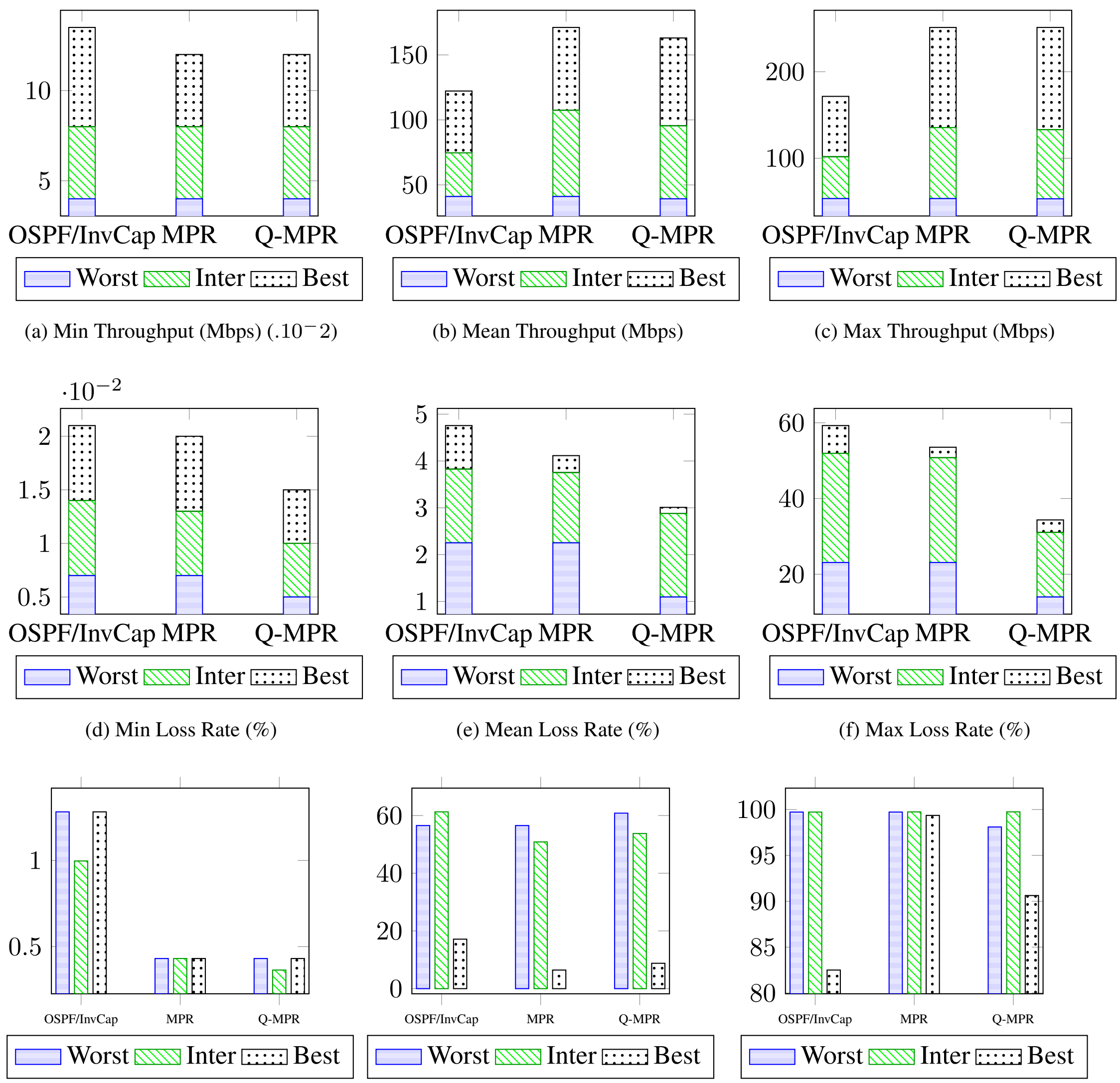
Figure 5 presents the performance of the four strategies regarding the total received throughput, the overall packet loss rate and the total session blocking rate for $X = 64$, value providing the best results in the offline algorithm (used as the network planning procedure). For each performance metric, we store and show the minimum, mean and maximum value throughout the simulation, for all GW-AR pair and for all planes, and in the worst, medium and best cast scenario (topology).

Open Shortest Path First protocol computes the routes towards all destinations in the network based on the shortest path in terms of number of hops. Link weights are typically set to 1, although a fixed constant different from 1 would produce the same result. The shortest-hop path is used between the gateway and each access router, regardless of the number of available paths towards the destination and the capacity of the path. Only one path will be used for forwarding traffic. With InvCap, which uses OSPF with an improved link weight setting, traffic towards a destination will still be routed along a single path. However, link weights are set to the inverse of the capacity of the link, that is a link with a low bandwidth will be penalized and assigned a high cost so that it will be avoided for path calculation. In other words, unless no other paths including this link is available, a link with a low capacity will be avoided. Traffic in InvCap therefore uses paths that are not necessarily shorter in terms of hop count, but more able to handle the amount of traffic. For lack of space, we could not present the limited differences in performance between OSPF and InvCap for these particular topology family (tree-like) but we noted the following. The minima are lower in InvCap than in OSPF. Maxima and mean values are identical, this is explained by the fact the topology are tree-like, and traffic is solely flowing between the gateway and access routers. As the network becomes overloaded, and because only path is available for each source-destination pair, the performance in both strategies is similar. It can clearly be noticed that, although InvCap offers better performance in transit or core networks compared to OSPF, it does not outperform OSPF in access networks. For these reasons, we decided to tie together OSPF and InvCap in the performance graphs in the rest of the paper.

With MPR, multiple routes are available between every GW-AR pair, as many routes as the number of routing planes. This has two consequences: first, traffic can be split over several paths, hence balancing the load within the network. This leads to increasing the overall throughput in the network and hence decreasing the blocking probability. Second, as shortest-hop routes are no longer used, higher delays are experienced by the sessions forwarded onto the RPs. In Q-MPR, for every new incoming session, the best plane, namely the best path, is selected for routing the session towards its destination based on the QoS requirements and the state of the plane. This will directly affect the blocking rate as more sessions will be denied access for lack of available paths. Blocking rate in Q-MPR is increased by 26% in the worst case compared to basic MPR strategy. Despite this effect, we denote that the overall throughput remains unchanged compared to MPR and presents a maximum gain of 45.2% compared to OSPF/InvCap schemes. It is explained by the fact that better paths are used for carrying traffic, the packet loss rate is lower in Q-MPR, with a maximum gain of above 75% compared to MPR and 85.9% compared to OSPF/InvCap. The end-to-end delay presents slightly lower values in Q-MPR compared to MPR in Topology 1, with a maximum gain of 61.6%.

In details, Fig. 5 depicts values for all metrics in a stacked-column structure, making it easy to compare performance across the studied approaches. Fig. 5(a),(b) and (c) show the total received throughput in Mbps; here the higher the value, the better. Looking at Fig. 5(b) , the mean throughput in the best scenario for Q-MPR (68 Mbps) is higher than that of MPR (63 Mbps), and OSPF/InvCap (47 Mbps). This becomes even more obvious by looking at Fig. 5(c). Fig. 5(d), (e) and (f) show the packet loss rate; here the smaller the value, the better performance. (d) and (f) show clearly that Q-MPR outperforms OSPF/InvCap and the QoS unaware MPR. In Fig. 5(f), Q-MPR in the intermediate sample topology presents a maximum loss rate of 17%, while MPR and OSPF/InvCap show higher values of 27% and 28% respectively. Looking at the minimum, average and maximum values enable us to assess the performance as not only the extreme values but also the median values are shown. Thus, one can draw a realistic picture of how the network is behaving.

Figure 6 and Figure 7 depict the performance of the studied strategies in the worst case, that is only one physical path is available for each GW-AR pair, in an intermediate case and in the best case, where the node degree is higher, as we increase progressively the total network load (normalized by the total network capacity). Q-MPR and MPR outperform OSPF and InvCap. For $X = 32$, Topology $T1M1$ presents the worst performance, OSPF/InvCap, MPR and Q-MPR perform similarly. Recall that in Topology $T1M1$ traffic can be routed only on one path. Therefore, only one routing plane is available in MPR and Q-MPR, downgrading their performance to that of OSPF and InvCap methods. The best case is shown with Topology $T1M3$, we note that with Q-MPR and MPR a higher amount of traffic can be carried in the network (see Fig. 6(g)). It can also be seen that OSPF/InvCap present a worse total packet loss rate and session blocking probability (see Fig. 6(e),(f),(h) and (i)). Finally, in Fig. 5(g), (h) and (i), the blocking probability, expressed in percentage, indicates the ratio of blocked sessions over the total number of incoming sessions. Q-MPR shows greater performance for the lower bound values but its performance decreases for average and maximum. OSPF



(a) Min Throughput (Mbps) (.10−2)
(b) Mean Throughput (Mbps)
(c) Max Throughput (Mbps)
(d) Min Loss Rate (%)
(e) Mean Loss Rate (%)
(f) Max Loss Rate (%)
(g) Min Blocking Rate(%)
(h) Mean Blocking Rate(%)
(i) Max Blocking Rate(%)

Figure 5: Total throughput, loss rate and blocking probability (min, mean, max) for worst, intermediate and best case scenarios. X = 64.

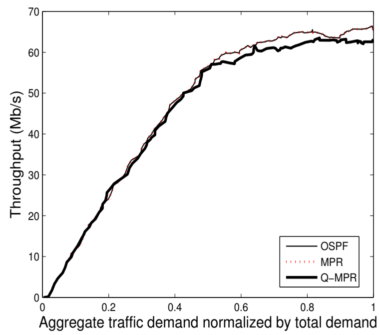
will tend to block sessions as routing engine will not find paths with sufficient bandwidths to route traffic, however Q-MPR will block sessions as it is more constrained. Hence the little gap in the blocking probability for average and maximum.

For $X = 64$, results are analyzed for the four strategies, and the worst case (Topology $T2M1$, Fig. 7(a),(b) and (c)), intermediate case (Topology $T2M4$, Fig. 7(d),(e) and (f)) and the best case (Topology $T1M4$, Fig. 7(g),(h) and (i)) are shown. From Fig. 7(h) and (i), it can be seen that Q-MPR perform better than its counterparts MPR, OSPF and InvCap for $X = 64$ than that of $X = 32$. Indeed, in Fig. 7(h), losses occur for a higher total traffic; 7% for Q-MPR with $X = 64$ against 3.5% with $X = 32$. Similarly, the total session blocking rate is slightly better in Q-MPR with $X = 64$ than with $X = 32$ (see Fig. 7(i) and Fig. 6(i)).

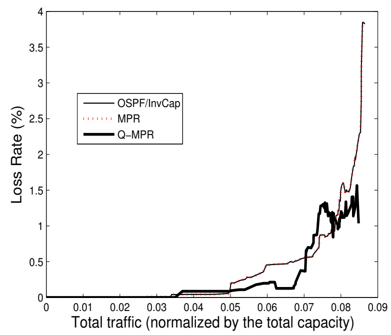
We demonstrated in this section that despite a fewer number of routing planes with $X = 64$ than with $X = 32$, better performance is achieved as more routing planes are tested in the offline algorithm, hence a better set of RPs can be found.

5. Conclusions

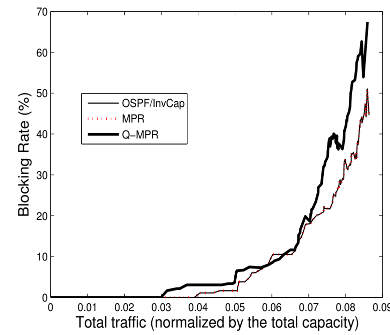
Building access networks' routing and traffic engineering via extensions proposed in the paper have shown to enable significant improvements in path diversity compared to standard IP routing. The extensions required in IP routers running OSPF for implementing the MPR method are comparable to the alternative solutions, both in the performance and flexibility. In the IP routing, ECMP could be used for comparison as it is typically applied in some types of topologies, but it is not able to flexibly accommodate for overall path diversity in a high degree of meshing and large number of nodes in IP access networks. On the other hand, MPLS is able to achieve path diversity but the high overhead of its installation and maintenance present a strong case for finding the solutions in IP routing adaptations as proposed in the paper. The paper additionally promotes IP access networks as a natural extension of the infrastructure of the Internet, not requiring additional networking support in the scoped segment of the network that provides access to wireless terminals. In addition, network planning and traffic engineering via QoS-awareness and the algorithms that comprise the MPR



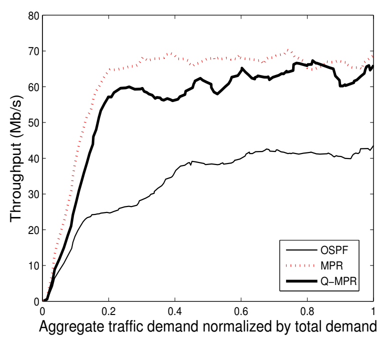
(a) T1M1



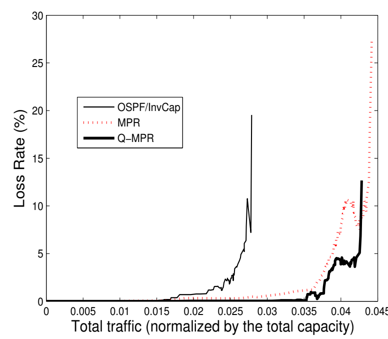
(b) T1M1



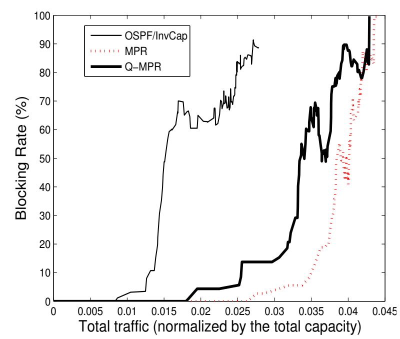
(c) T1M1



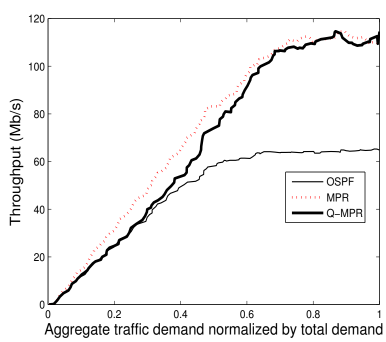
(d) T2M3



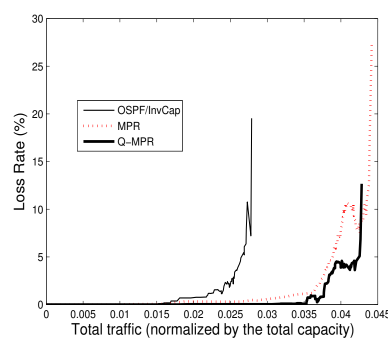
(e) T2M3



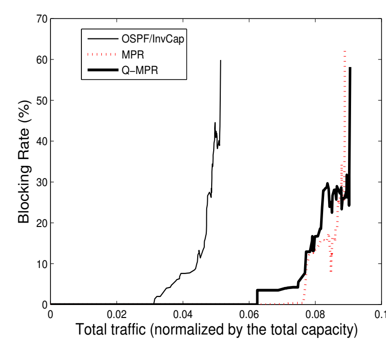
(f) T2M3



(g) T1M3

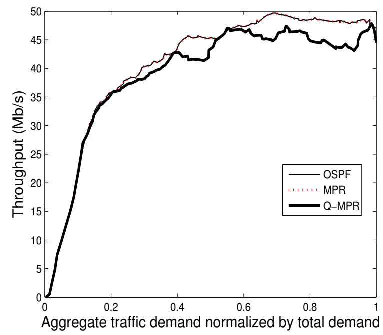


(h) T1M3

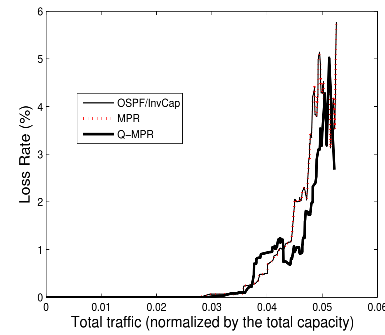


(i) T1M3

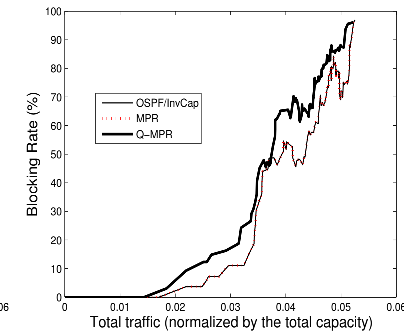
Figure 6: Total received throughput, packet loss rate and session blocking rate, X = 32.



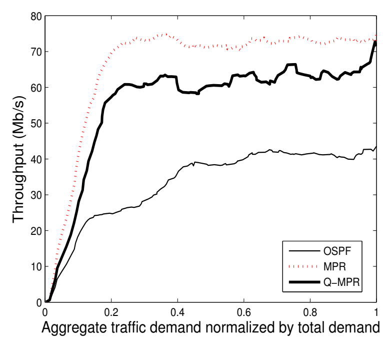
(a) T2M1



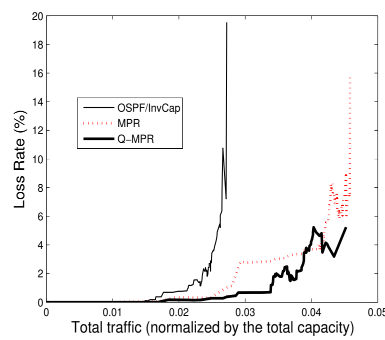
(b) T2M1



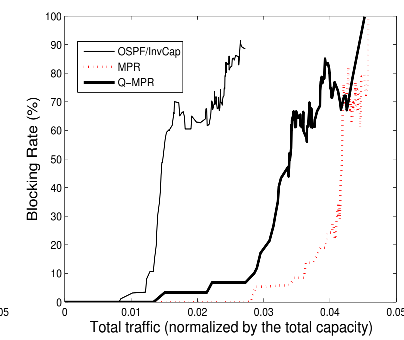
(c) T2M1



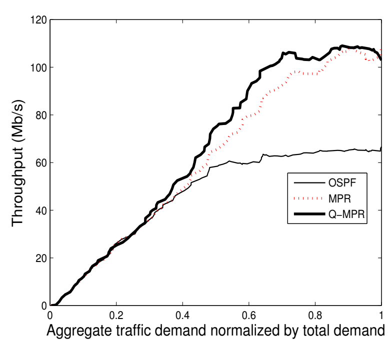
(d) T2M4



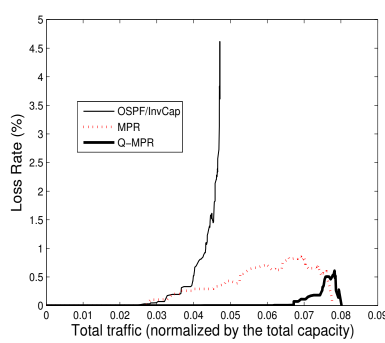
(e) T2M4



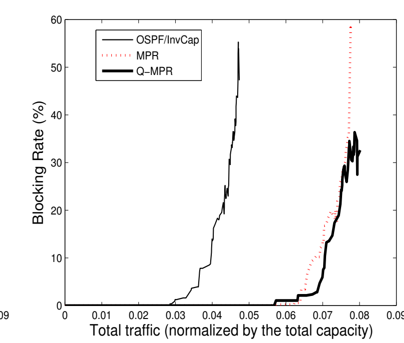
(f) T2M4



(g) T1M4



(h) T1M4



(i) T1M4

Figure 7: Total received throughput, packet loss rate and session blocking rate, X = 64.

method are features that would equally be needed for other networking solutions in IP access networks, e.g. MPLS already uses OSPF for LPS path computations and traffic engineering. QoS-aware MPR allows the network to maintain several independent logical topologies that can be used to balance the traffic load within the network whilst providing QoS for end users. Our method classifies new incoming sessions and routes them at the edge of the network, namely at the gateway and at the access nodes, onto the routing plane that achieves best network performance and that provides best QoS for the user. The method uses both an offline and an online process for network planning and traffic engineering respectively, and the performance issues were addressed both theoretically and by simulation. The results showed clearly our Q-MPR scheme outperforms existing strategies even with a small number of routing planes (5 for high-meshed access networks). Using Matlab and the NS-2 simulator, we compared Q-MPR against basic MPR, OSPF and the InvCap mechanisms.Total received throughput is increased by 45.2% with MPR compared to OSPF and InvCap strategies. Q-MPR, while generally blocking more sessions and using the same routing plane configuration as that of MPR, achieved the same overall throughput whilst lowering the total packet loss rate. For future work, we would like to investigate the portability of our method to other topologies and network models, e.g. wireless mesh networks, multiple gateways and various combinations of source/sink nodes in IP access network. Finally, we will investigate offering MPR as a candidate solution for SDN.

Acknowledgment

This work was made possible thanks to the help of two Masters visiting exchange students, Paul Faucheux and Matthieu Danel, who I was honored to supervise and contribute greatly to this paper. This work was funded by the EPSRC.

References

[1] J. Fabini, P. Reichl and A. Poropatich,, *"A generic approach to access network modeling for next generation network applications"*, in Proc. 4th International Conference on Networking and Services (ICNS), March 2008.

[2] L. Muscariello, D. Perino and D. Rossi, *"Do next generation networks need path diversity?"*, in Proc. International Conference on Communications (ICC), June 2009.

[3] C. Keszei, N. Georganopoulos, Z. Tyranyi and A. Valko, *"Evaluation of the BRAIN candidate mobility management protocol"*, in IST Mobile Summit, Germany, 2001.

[4] S. Gundavelli, K. Leung, *"Proxy Mobile IPv6"*, in RFC 5213, August 2008.

[5] A. Jaron, P. Pangalos, A. Mihailovic and H. Aghvami, *"Proactive autonomic load uniformisation with mobility management for wireless Internet Protocol (IP) access networks"*, in IET Networks, vol.1, no.4, pp.229,238, December 2012.

[6] R. Callon, A. Viswanathan and E. Rosen, *"Multiprotocol label switching architecture"*, IETF, RFC 3031, 1999.

[7] B. Fortz and M. Thorup, *"Internet traffic engineering by optimizing OSPF weights"*, in Proc. of 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), vol.2, pp.519-528, 2000.

[8] A. Kvalbein and O. Lysne, *"How can multi-topology routing be used for intradomain traffic engineering"*, in Proc. of IEEE SIGCOMM Workshop, 2007

[9] N. Wang, K. Ho and G. Pavlou, *"Adaptive multi-topology IGP based traffic engineering with near-optimal network performance"*, in Proc. of IFIP International Federation for Information Processing, NETWORKING, pp.645-66, 2008.

[10] P. Pesnak, L. Nguyen, S. Mirtorabi, P. Pillay-Esnault and A. Roy, *"MT-OSPF: Multi-Topology (MT) routing in OSPF"*, IETF, RFC 4915, 2007.

[11] J. Wang, Y. Yang, L. Xiao and K. Nahrstedt, *"Path splicing"*, Computer Network, Elsevier, vol.48, no.4, 2005.

[12] M. Motiwala, M. Elmore, N. Feamster, and S. Vempala, *"Edge-based traffic engineering for OSPF networks"*, In Proceedings of the ACM SIGCOMM 2008, USA.

[13] A. Kvalbein, A.F Hansen, T. Cicic, S. Gjessing and O. Lysne, *"Multiple Routing Configurations for Fast IP Network Recovery"*, Networking, IEEE/ACM Transactions on , vol.17, no.2, pp.473,486, April 2009.

[14] A. Jaron, A. Mihailovic and A.H. Aghvami, *"Introducing Multi-Plane Routing for next-generation all-IP wireless access networks"*, in Proc. of the International Conference on Communication (ICC), 2012.

[15] A. Jaron, M. Danel, P. Faucheux, A. Mihailovic and A.H. Aghvami, *"QoS-aware Multi-Plane Routing for future IP-based access networks"*, in Proc. of the Global Communications Conference (GlobeCom), 2012.

[16] Z. Wang and J. Crowcroft, *"Bandwidth-delay based routing algorithms"*, in Proc. of the GLOBECOM, vol. 3, pp.2129-2133, November 1995.

[17] T. Korkmaz and M. Krunz, *"Multi-constrained optimal path selection"*, in Proc. of the IEEE INFOCOM, vol. 2, pp.834-843, April 2001.

[18] Cisco, *"Policy-based routing"*, White paper, 1996.

[19] A. Jaron, P. Faucheux, M. Danel, A. Mihailovic and A.H. Aghvami, *"Implementing extensions to IP routing for increasing throughput in Future Internet"*, Wireless World Research Forum (WWRF) Meeting 28, April 2012, to appear.

[20] A. Jaron, M. Danel, P. Faucheux, A. Mihailovic and A.H. Aghvami, *"Multi-Plane Routing, fitting IP routing in modern access networks"*, http://www.ctr.kcl.ac.uk/MPR/.

[21] S. Blake *et al.*, *"An Architecture for Differentiated Services"*, RFC 2475.

[22] F. Baker, D. Black, S. Blake and K. Nichols, *"Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers"*, RFC 2474.