Comparison of Multi-service Routing Strategies for IP Core Networks

Ulf Jensen and Armin Heindl

Universität Erlangen-Nürnberg
Computer Networks and Communication Systems
Erlangen, Germany

Abstract. Service differentiation in IP core networks may be supported by dedicated path selection rules. This paper investigates the degree of service distinction achievable when common routing strategies, like ECMP, SWP and WSP, are applied to two traffic classes separately and in different combinations. One traffic class requires low latencies, while the other is considered as best-effort traffic.

A Maple program has been developed that evaluates network performance characteristics, like maximal link utilization, and per-class measures, like mean end-to-end delay and mean number of hops, when paths are computed on demand with traffic demands arriving in arbitrary order. Realistic network topologies may be imported from the publicly available tool BRITE, while link capacities and traffic patterns are chosen randomly (with realistic constraints) in Maple.

Experiments show that a comparable service differentiation may already be achieved with less sophisticated strategy combinations, which apply ECMP to the delay-critical traffic class.

1 Introduction

New applications and services [1] in the next-generation Internet (NGI) require different service guarantees typically negotiated in Service Level Agreements (SLA). In order to provide appropriate service differentiation in the Internet, many proposals have been made including diverse strategies for packet classification, queue management and scheduling as well as bandwidth management and admission control. For instance, such issues are addressed in the service models for Quality of Service (QoS) support by the Internet Engineering Task Force (IETF), namely Differentiated Services (DiffServ, [2]) and Integrated Services (IntServ, [3]). Today, these technologies coexist with other approaches to provide QoS. Traffic engineering capabilities are supplied by Multi-Protocol Label Switching (MPLS, [4,5]) and QoS routing constitutes another important component in the overall QoS framework [6,7]. Architectures like GMPLS (Generalized MPLS, [8,9]) and extensions to common IP routing protocols, like OSPF (Open Shortest Path First, [10]), furnish the tools to handle traffic classes according to different rules, but it is not fully understood to which extent such routing decisions contribute to service differentiation or which rule combinations result in favorable performance.

In this paper, we investigate these issues for IP core networks. We assume that services may be set up on a semi-permanent basis, i.e., a negotiated bandwidth has to be

K. Al-Begain, D. Fiems, and G. Horváth (Eds.): ASMTA 2009, LNCS 5513, pp. 1–15, 2009.

reserved along a path or multiple paths through the network for a longer period. Service requests are served on demand, i.e., the routes cannot be optimized from a global perspective (in the knowledge of all traffic demands), but have to be computed incrementally upon arrival of a request. Such a scenario is typically encountered in traffic engineering, and its solution easily realized, e.g., by means of label-switched paths as in (G)MPLS. Furthermore, we consider two traffic classes: one traffic class is related to interactive applications and requires lower latencies, while the other represents best-effort traffic. Different routing strategies are applied to each class: we distinguish alternative link weight systems, single/multi-path routing as well as standard/QoS routing schemes, which disregard/regard the current state of the network.

A Maple program has been developed to read arbitrary topologies of realistic sizes generated with the publicly available tool BRITE [11], to assign link capacities, to generate traffic patterns for the two traffic classes and to allocate the traffic demands to the network according to specified rules. Finally, the overall network peformance is assessed by means of the maximal link utilization and the minimal unused link capacity. The service differentiation is ascertained in terms of the per-class performance measures, like the average number of hops along the paths and the mean end-to-end delays.

Just as Internet routing itself is focused on connectivity with QoS having been addressed much later, studies on traffic engineering are primarily targeted on issues like load balancing and improvement of overall network performance instead of service differentiation (e.g, [12,13,14,15,16]). While sophisticated QoS routing architectures have been proposed (e.g., [17,18]) and mostly been evaluated by discrete-event simulation on the packet level [18,19], many questions regarding fundamental design decisions remain open. In the context of Internet backbone networks, we are interested inasmuch basic routing schemes cooperate or interfere when applied to different traffic classes.

Our approach has been inspired by a case study in [15], which examines similar constraint-based routing schemes in the context of traffic engineering for a single traffic class. We extend Wang's procedure of incremental demand assignment to two traffic classes in order to study the potential of service differentiation.

The paper is organized as follows: Section 2 presents the routing schemes, which are applied in our experiments. The experiment setup and model evaluation is described in Section 3, while numerical results in Section 4 compare different routing combination for service differentiation. The paper concludes with Section 5.

2 Considered Routing Strategies

Traditional routing in the Internet is based on shortest paths between origin and destination. Each router (being the origin) computes this shortest path locally based on its view of the network. The decision to which output interface the packet is directed depends solely on the destination of the packet, which implies that all types of traffic are equally processed. In order to enable service differentiation between best-effort traffic and higher-priority traffic, current routing protocols, like OSPF [10,20], and architectures, like GMPLS [8], make provisions to manage separate routing tables for different traffic classes, which are computed based on different link metrics. Furthermore, path computation may not only be based on metric weights assigned a priori and independently of the dynamic network state, but may also reflect traffic engineering information like the unreserved bandwidth by priority, etc. Thus, various variants of constraint-based path computation may be realized besides standard link-metric routing.

Commonly, routers periodically exchange network state information so that each one can calculate the best path(s) based on some knowledge of the entire network state conditions [17]. Other strategies have been proposed, e.g., using local information to choose a path from predefined candidate paths [21], but remote network conditions prove crucial to QoS routing. We follow the principle approach suggested in [17] that fits into the GMPLS architecture and is thus highly compatible with the existing and widely deployed routing protocols. In this paper, we focus on standard link-metric routing and QoS routing strategies solving the bandwidth-restricted path (BRP) problem [6] (via metric ordering), namely Shortest-Widest-Path (SWP) and Widest-Shortest-Path (WSP). In the context of QoS routing, we do not consider solutions to the so-called Restricted-Shortest-Path problems due to their higher complexity and neither approaches of metric combination due to their rough heuristic. Generally, QoS routing may suffer from computation and communication overhead, which hamper scalability, or from a strong snsitivity on the view of the network, where inaccuracies strongly degrade performance. These issues are only addressed in this paper in the sense that we concentrate on less sophisticated QoS routing strategies.

2.1 Standard Link-Metric Routing

Here, the link weights are set to static default values and independently of the network dynamics. In order to compute the shortest path to a destination, each router may apply classic Dijkstra algorithms [22]. Naturally, the length or cost of a path is the sum of all weights on the links between origin and destination.

Minimal Number of Hops (MH). Still today, the unit metric system is applied in large parts of the Internet. All link weights are set to 1. The shortest path minimizes the number of hops along the way between origin and destination.

Inverse Link Ratio (ILR). This metric system reflects the static link capacity. The link weight is set to the reciprocal value of the link capacity. Thus, links with high capacity attain smaller weights and are thus favored in the shortest path computation based on these weights. The rationale that traffic travels faster on high-capacity links may, however, be counteracted by attracting high loads to these links. This metric system has become very popular as a default setting in today's routers.

Equal-Cost Multi-Path (ECMP). While the above strategies are associated with single-path routing, the add-on property ECMP allows to route traffic from origin to destination on multiple shortest paths. Essentially, the ECMP rule states that – if multiple shortest paths exist – a flow is split equally. More precisely, "a flow to a destination outgoing from a node is equally split onto these outgoing links which belong to the shortest paths to this destination." (from [22], where a recursive ECMP flow allocation algorithm is given). ECMP is realized with minimal modifications of the routing tables, which now contain a next hop for every shortest path to a destination. All (or some) shortest paths can be obtained by means of the k-shortest-path-algorithm based on any arbitrary metric, like MH, ILR, etc. ECMP balances the network load and is an important feature of OSPF with impact on traffic engineering.

2.2 QoS Routing

QoS routing finds an optimal path that satisfies a particular request under constraints, which reflect the dynamic state of the network. As a common example, the widest path maximizes the so-called bottleneck bandwidth between origin and destination [23]. The bottleneck bandwidth represents the minimal unused capacity of all links along a path. Obviously, routers must therefore exchange information on the unreserved bandwidth of links. Widest paths are well suited for load balancing, since paths with higher remaining capacities are preferred. Longer paths in terms of number of hops, however, strain the overall utilization of the network. More hops may also induce longer delays.

The Dijkstra algorithms for shortest paths are straightforwardly adapted to widest-path computations [22]. The specific changes required for the algorithms applied in this paper can be found in [24].

Since both metrics – a minimal number of hops and a high unused capacity along the path – have their benefits, paths are desired which combine these favorable properties to some extent. Solving the related bandwidth-restricted path problem implies the heuristic of metric ordering, i.e., first the best paths are found with respect to one metric and then – among these best paths – the best path with respect to the other metric is determined.

Widest Shortest Path (WSP). WSP algorithms first determine all shortest paths in terms of a standard metric (independent of network load), between which the tie is broken via the largest bottleneck bandwidth. Especially when MH is used in the first step, this metric ordering emphasizes low resource consumption in the network. WSP is computationally efficient, works well also for high network loads and/or with inaccurate network info.

Shortest Widest Path (SWP). SWP algorithms turn the metric ordering around: among all widest paths (possibly from a candidate list), the shortest one (according to a standard metric, like MH or ILR) is eventually selected. Determining widest paths first results in eventually longer paths. SWP primarily aims at load balancing. It scales well, especially in combination with path precomputation, but exhibits a more selfish behavior penalizing later requests.

In the next section, we describe in which setting these routing strategies are applied to two traffic classes in different combinations. Traffic demands of these classes incrementally routed over an initially empty network. All algorithms and computations have been implemented as Maple procedures [25].

3 Experiment Setup and Model Evaluation

Section 3.1 describes the chosen experiment setup while Section 3.2 addresses how a Maple program is used to evaluate the experiments.

3.1 Experiment Setup

An experiment comprises five steps: topology generation and read-in, traffic creation, path determination, traffic allocation and performance measure. Thereby, each traffic class uses a preassigned strategy for path determination.

Topology Generation and Read-In. At first, the tool BRITE [11] is used to generate realistic but artificial backbone topologies. In BRITE, the user can choose from a range of network models to create topologies. The employment of the Barabási-Albert model creates topologies with a majority of nodes of low degrees. The degree of a node reflects the links connected to it. Link capacities are chosen according to a discrete uniform distribution using the first seven levels of the European multiplex hierarchy. Hence, capacity values range from 51.84 Mbps to 1866.24 Mbps, which represent the lower and upper bound of the chosen multiplex levels. The links are bidirectional and share their bandwidth as needed.

Traffic Creation. The presented software produces dedicated traffic flows on traffic creation. A flow from source s to destination t will be distinguished from a flow from source t to destination s. Prior to traffic demand creation, the nodes are split in boundary nodes and transit nodes. Traffic demands use boundary nodes as source and destination, transit nodes are only used as intermediate nodes. The partitioning into boundary and transit nodes is carried out considering the degree of every node and the average node degree. For details of this procedure see [24]. Once the nodes are partitioned, static traffic demands can be established. A traffic volume drawn from a uniform distribution in an arbitrary, but fixed interval is assigned to each demand. The traffic generation process is the same, no matter how many traffic classes are used. To conclude this step it can be stated that after splitting in transit and boundary nodes, the desired amount of dedicated demands are established between unique pairs of boundary nodes.

Path Determination and Traffic Allocation. Traffic classes may assume different shares of the overall traffic. This means that each class has a fraction of demands from the set of overall demands. For example, if the best-effort class and the delay-critical class have the same portion of traffic and ten traffic demands are to be allocated, each class has to realize five demands. Starting with the total amount of estimated traffic demands, each demand is assigned to a traffic class as follows.

1. A demand is randomly chosen.
2. The traffic class that will realize this demand is chosen. If both classes still require demands, one class is randomly picked. Otherwise, the demand will be realized by the only class which still has traffic to carry. This procedure ensures a ”mixing” during the allocation of demands of the different traffic classes.
3. Having chosen a traffic class the demand belongs to, the routing path needs to be determined using the strategies described in Section 2. In case of multi-path routing (e.g. ECMP), potentially more than one path has to be identified.
4. The allocation of traffic simply means substracting the demand volume from the capacities of all links along the routing path(s).

With this procedure, the network capacity will be downsized step by step with every allocated demand.

Performance Measures. In the final experiment step, performance measures are computed. We first discuss network measures, which assess the aggregate performance of the routing, and then per-class measures, which manifest the service differentiation between the traffic classes. Having saved the idle topology (i.e., with the original capacities fully available), it is easy to determine network characteristics using the loaded and idle network. The maximal link utilization and the minimal unused link capacity are determined. The formula

$$U = \max\left\{\frac{y_e}{c_e} : e \in E\right\}$$

purveys the maximal link utilization in percent, where E denotes the set of edges/links and c_e their capacity. The term

$$y_e = \sum_d \sum_{p=1}^{P_d} \delta_{edp} x_{dp}$$

describes the traffic on link e. The variable x_{dp} contains the demand volume that is routed on path p of the P_d paths which realize demand d. The logical value δ_{edp} determines whether path p (realizing demand d) uses link e ($\delta_{edp} = 1$) or not ($\delta_{edp} = 0$). The minimal unused link capacity is computed in Mbps by using the formula

$$C = min\left\{c_e - y_e : e \in E\right\} \quad .$$

To calculate traffic characteristics for each traffic class, the routing paths for each traffic demand are needed. This data, together with the idle and loaded network, is the base to compute the average number of hops on the path and the mean path delay. The mean path delay depends on the link delay of each link on the path. The link delay is approximated with the formula

$$D_e = \frac{1}{c_e - y_e} \quad ,$$

in analogy to the time in the system of an M/M/1-queue. This rather rough approximation is sufficient for our purposes of a relative comparison. The mean path delay, weighted with the demand volume, is computed using the formula

$$D = \sum_d \sum_{p=1}^{P_d} \frac{x_{dp}}{\sum_d h_d} \sum_e \delta_{edp} \frac{1}{c_e - y_e} \quad . \tag{1}$$

The variable h_d represents the demand volume of demand d. This delay includes transmission and queueing delay and is denoted in seconds. Another important traffic characteristic is the average number of hops which is also needed to compute the processing delay. The formula

$$H = \sum_d \sum_{p=1}^{P_d} \frac{x_{dp}}{\sum_d h_d} \sum_e \delta_{edp}$$

is used to compute this characteristic. The paths are again weighted according to their demand size. To estimate the end-to-end delay of the traffic of a traffic class, presumptions about the propagation delay and the processing delay per hop have to be made.

This means, that both traffic characteristics shall be contemplated together with network presumptions for the propagation and processing delay.

3.2 Usage of Maple Software

Executing the above-mentioned five steps results in network and traffic characteristics for a fixed amount of demands and a specified demand ratio of the traffic classes. To extensively evaluate the performance of two routing strategies, we obtain values for different amounts of demands. The use of different demand ratios is of high practical interest. In addition, the randomness in traffic generation and demand assignment in an experiment is coped with by executing an adequate amount of repetitions. The developed Maple program provides for independent replications. Comprehensive experiments are executed with a single function call, while the architecture allows simple enhancement and adaption. The following section deals with usage, prospects and adaption options of the software.

An experiment to compare two routing strategies that are assigned to a traffic class each consists of different partial experiments. In these partial experiments, the portion of demands of each traffic class on the overall demand amount differs. The results of Section 4 use the ratios presented in Table 1. The ratios can easily be adapted to satisfy practical situations.

Table 1. Share of demands of each traffic class of the overall demand amount

	partial experiment			
traffic class	1	2	3	4
best effort	100%	90 %	70%	50%
delay critical	0%	10 %	30%	50%

For the traffic classes, every desired combination of routing strategies can be chosen. The strategies described in Section 2 are just a selection of common policies. Other strategies can be easily added and are specified as an argument when calling the experiment function which executes automatically the partial experiments. Another argument is the ratio of boundary and transit nodes. Sufficiently many boundary nodes are needed to create more unique source-destination-pairs as traffic demands are specified. The interval for the demand volumes is an argument, too. The number of repetitions may be specified as a tradeoff between statistical significance and execution time.

The developed program offers the opportunity of executing sophisticated experiments that deliver extensive results while keeping usage easy. Numerous arguments allow simple adaption and the modular architecture ensures easy enhancement.

4 Comparison of Multi-service Routing Schemes

This section presents results of experiments conducted with a topology of 40 nodes and 77 links (as generated with BRITE). The link capacities are uniformly chosen from the first seven layers of the European multiplex hierarchy. Further preferences are an interval between 3 and 12 for the demand volumes and the portion of 0.9 of all nodes for the boundary nodes. The experiment starts with 20 demands and increases its number by 10 up to 130. For the latter traffic, a maximum link utilization of around 60% is attained for every experiment, which is considered as a reasonable operational load.

The result figures for the network performance characteristics show one curve for each partial experiment (see Table 1). The solid curve displays the behavior without traffic differentiation. In this case, the whole traffic is routed using the best-effort strategy. The dashed line represents a share of 10% delay-critical traffic of the overall demands. Partial experiment 3, illustrated in the dashed-dotted line, has a share of 30% delay-critical traffic while the dotted line shows the results of the experiment where both traffic classes realize the same number of demands.

One partial experiment is chosen to display the per-class traffic characteristics. The figures show partial experiment 3 with a share of 30% of the overall demands for the delay-critical traffic class. For comparison, the traffic characteristics of partial experiment 1 with no traffic differentiation are shown as well.

To produce the results of each subsection (with seven replications in each specific setting, i.e., for each demand number in each partial experiment), run times on standard PCs varied between one hour for the simpler strategies and some hours for more complex experiments with QoS routing. We do not show confidence intervals for our results in order not to overload the figures. They can be found in [24].

Among the various strategies we have evaluated, we show here three routing combinations which highlight the potential of service differentiation without and with QoS routing strategies.

4.1 ECMP (ILR) for Best Effort and ECMP (MH) for Delay-Critical Traffic

This combination was selected to find out whether strategies that do not use QoS routing techniques are able to provide service differentiation by different weight assignments. ECMP with the MH strategy is selected for the delay-critical traffic, since shortest paths with the minimal number of hops minimize the per-node delays, while link loads are somewhat balanced by ECMP leading to lower uilizations and thus lower delays. Note that ECMP splits a demand over different paths of the same length. Best-effort traffic is merely engineered to balance the network load. Besides the ECMP multi-path routing, the ILR link weight assignment tends to direct traffic over possibly longer paths with links of higher capacity. The static weights of inverse link capacities allow to avoid bottlenecks slightly better than unit weights.

Figure 1 displays the network performance characteristics. According to intuition, as the load on the network increases, the maximal link utilization increases and the minimal unused link capacity decreases. With respect to the different partial experiments, we only discuss the maximal link utilization. In partial experiment 1 (solid line), all traffic is routed according to ECMP (ILR), i.e., without any traffic differentiation. Compared with the other partial experiments with two traffic classes, this weight assignment due to link capacities yields benefits in low network loads (see lower maximal link utilization), but tends to show worse network performance for higher loads. In our experiment setting, the maximal link utilization reaches values of beyond 80 %. Also with increasing load, ILR prefers high-capacity links, which become heavily loaded. This effect is

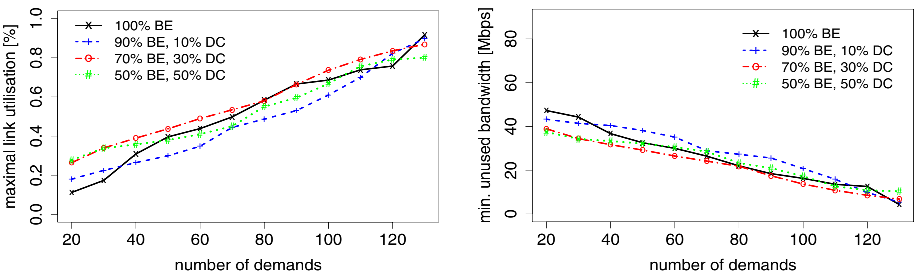

Fig. 1. Network characteristics for experiment with ECMP (ILR) for best-effort class and ECMP (MH) for delay-critical class: maximal link utilization is shown left, minimal unused link capacity on the right for different partial experiments

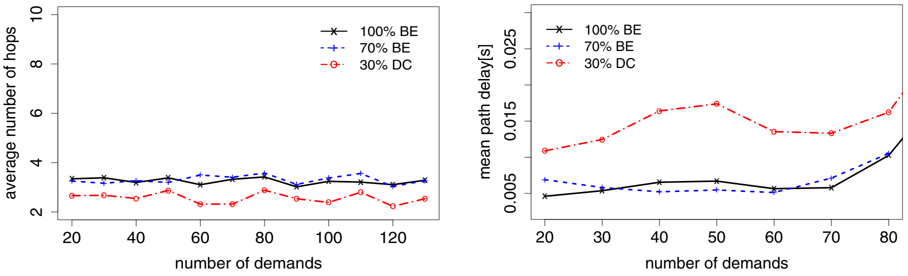

Fig. 2. Traffic characteristics for experiment with ECMP (ILR) for best-effort class and ECMP (MH) for delay-critical class: average hop count is shown left, mean path delay on the right for partial experiments 1 and 3

mitigated in the presence of delay-critical traffic, which is routed according to different rules, which explains the relatively bad performance of partial experiment 1.

In the center range, best network performance is achieved for the partial experiment 2 with a 10 % share of delay-critical traffic (see dashed curve). With higher such shares, network performance deteriorates again, even more for 30 % (dashed-dotted line) than for 50 % (dotted line). Obviously, small shares (around 10%) of delay-critical traffic may have a positive influence on the network performance. We assume that the load on lower-capacity links is too small to influence the overall network performance in partial experiment 2.

The traffic characteristics per class are shown in Figure 2 for partial experiment 3 (30 % share of delay-critical traffic) along with reference curves for partial experiment 1. Considering the average hop count (left), the curves reveal that the delay-critical class traffic (dashed-dotted line) is routed on shorter paths as desired to reduce the per-node overhead. The solid line of partial experiment 1 (100 % best-effort traffic) and the dashed line of the best-effort share are quite similar, with the dashed line assuming slightly larger values in most cases. A difference of one hop (on average) can be observed between best-effort and delay-critical traffic and substantiates the presumptions for choosing this strategy combination. The MH weight assignment accounts for shorter routing paths, while ILR pays the price for avoiding low-capacity links with longer paths. The paths of the best-effort class are even longer than the paths of the partial experiment 1 without traffic differentiation, because bottleneck links caused by the delay-critical class need to be avoided additionally. A small but considerable differentiation can be stated for the traffic characteristic of the average hop count.

The curves for the mean path delay are shown up to an amount of 80 demands. Above this number, meaningful results could not be obtained due to overload situations. With the maximal link utilization (averaged over 7 replications) beyond 60 %, the probability that a single link in one of the replications becomes overloaded increases. In such a case, the mean path delay (see equation (1)) can no longer be computed [24].

The curve constellation in Figure 2 (right-hand side) showing the mean path delay is counterintuitive at first sight. The dashed-dotted line of the delay-critical class ranks above the best-effort curve meaning that a best-effort traffic is routed on paths with lower mean delays. Since the delay-critical traffic uses fewer hops on average, it must traverse links which are more heavily loaded. According to (1), the link delays are the crucial factor. Nevertheless, another issue needs to be adressed. The path delay formula considers only transmission and queueing delays and does not regard propagation and processing delays. For mean end-to-end delays, propagation and processing delays have to be considered. Then, both traffic characteristics have to be evaluated together, because processing delays are added for every hop and propagation delays for every link used on the routing path. An assumption of 10 to 15 ms per hop for processing and propagation delay is reasonable for the considered IP core networks and leads to a different situation for the evaluation of the end-to-end delay for the different traffic classes. A difference of 10 ms on average for the mean path delay and one hop on average for the hop count (as roughly shown by Figure 2) leads to the conclusion that the traffic will only be differentiated with respect to the mean end-to-end delays, if the processing and propagation delays rise above 10 ms. That means, to make clear predictions, details about the network topology, like distances between nodes and node behavior, need to be known. The strategy combination considered in this subsection is only useful for traffic differentiation (for the mean end-to-end delays), if processing and propagation delays are considerably larger than 10 ms.

4.2 SWP (MH) for Best-Effort and ECMP (ILR) for Delay-Critical Traffic

This experiment applies the QoS routing strategy SWP to the best-effort class to further exploit load balancing for this traffic class. The MH weight assignment in SWP finds the shortest path (in terms of number of hops) of a set of widest paths. Since widest paths take into account the dynamic state of the network, load balancing is improved. The predominant IP routing strategy ECMP with weight determination ILR is used for the delay-critical class and is based on static network properties.

As before, Figure 3 displays the network characteristics and shows similar reciprocal trends for both network performance characteristics. However, with respect to the previous experiment, the network performance is now considerably improved: for 130 demands, the maximal link utilization is now around 60 % (as opposed to 80 %) and the minimal unused link capacity remains over 20 Mbps (as opposed to less than 10 Mbps). As a consequence, all curves show smaller slopes than in the previous experiment.

The QoS routing strategy for the best-effort class made load balancing more effective and stable. All curves in each figure are quite close to each other, where the partial experiments with a higher share of delay-critical traffic (dotted and dashed-dotted lines) show slightly worse network performance. For higher loads, the ordering of the partial experiments is as expected: with an increasing share of the delay-critical traffic (with non QoS routing in this experiment), the network performance deteriorates. The good performance of partial experiment 1 (10 % delay-critical traffic, dashed line) in the center part may be attributed to the suitable ILR weight assignment for low network load.

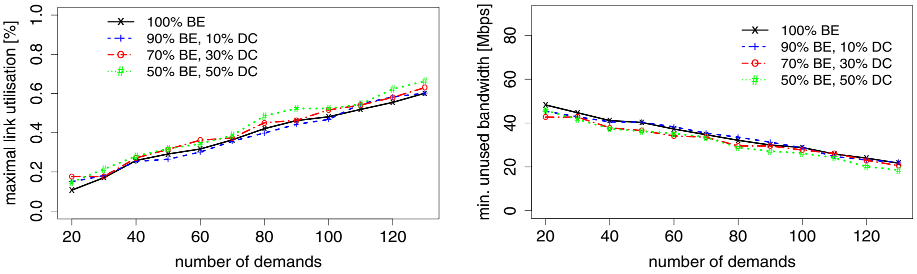

Fig. 3. Network characteristics for experiment with SWP (MH) for best-effort class and ECMP (ILR) for delay-critical class: maximal link utilization is shown left, minimal unused link capacity on the right for different partial experiments

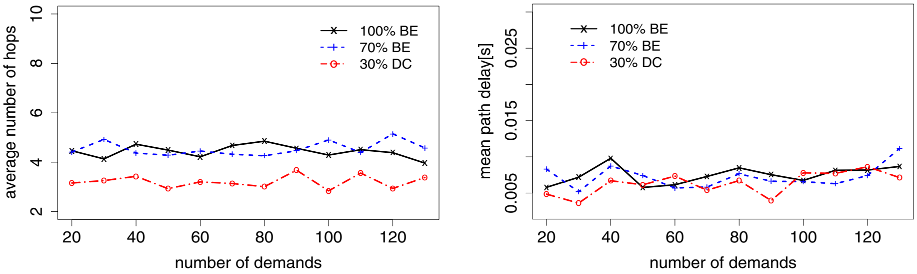

Fig. 4. Traffic characteristics for experiment with SWP (MH) for best-effort class and ECMP (ILR) for delay-critical class: average hop count is shown left, mean path delay on the right for partial experiments 1 and 3

Figure 4 shows the per-class traffic characteristics. A significant differentiation can be observed. With respect to the mean number of hops (see left-hand side), the quantitative difference, estimated to about one and a half hops, is larger than the one in the previous experiment. The values for ECMP (ILR) remained rather unchanged, while SWP (MH) uses paths that contain more hops on average. This use of widest paths results in a considerable differentiation of the traffic classes for this characteristic.

The second traffic characteristic on the right side of Figure 4 appears quite different compared with the experiment of Section 4.1. Due to the well-balanced network load, meaningful results could be achieved for demand numbers up to 130. In addition, the curve constellation itself is remarkable as both traffic classes reach quite similar values.

The dashed-dotted line for the delay-critical traffic lies below the dotted line for the best-effort traffic for most of the calculated values. As intended, the best-effort class which uses longer paths on less loaded links now appears to encounter longer delays than the delay-critical class, with uses potentially shorter paths with high capacities. Considering both traffic characteristics together and assuming the same 10 ms for processing and propagation delay as in the previous experiment, we conclude that a significant service differentiation can be reached with respect to mean end-to-end delays with the strategy combination chosen for this experiment. The delay-critical class traffic will be routed faster even without taking topology details into account. The performance benefit will further increase with higher values for the processing and propagation delay.

Further experiments with the same strategy combination were accomplished to back up the observed results. On the one hand, the topology size was raised while using the same network generation model. On the other hand, another generation model, the Waxman model [11], was utilized in BRITE while keeping the network size constant. Other preferences as described in Section 3.1 were not changed.

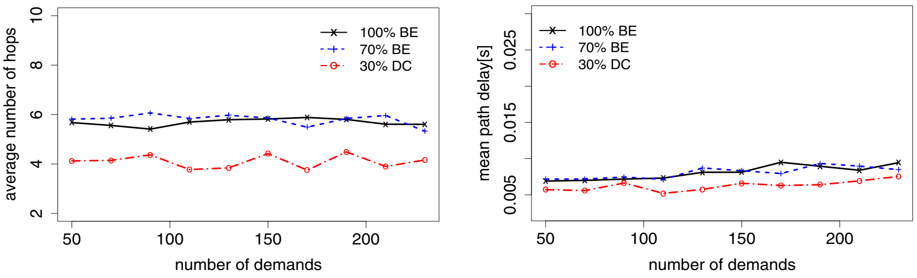

Fig. 5. Traffic characteristics for 50 node experiment with SWP (MH) for best-effort class and ECMP (ILR) for delay-critical class: average hop count is shown left, mean path delay on the right for partial experiments 1 and 3

Figure 5 shows the traffic characteristics for the experiment with a 50 node topology and same network generation model usage. The results for the application of the Waxman model show the same trends and can be extracted from [24]. The figures reveal, that the number of demands was increased notedly. In a larger network, more traffic is needed to reach a adequate operation load. With the higher amount of demands, the curves are mainly flattened as the randomness in traffic generation and demand assignment is balanced.

The average hop count curves that appear on the left side show a significant differentiation with a difference of two hops in average. The values for all curves increased with the network size as there are simply longer paths needed on the way from source to destination. The best-effort class (dashed line) uses six hops while the delay-critical class needs four hops on the path. The increased difference in comparison to the 40 node network experiment can also be imputed to the larger topology.

Contemplating the right figure with the mean path delay, it can be stated that the dashed-dotted line (delay-critical class) lies below the dashed line of the best-effort class. SWP, as used for the best-effort class, uses more hops and adds path delay for each link utilized. Although these links are few loaded, WSP that uses less but potentially higher loaded links finds paths with less queuing and transmission delay in average. This differentiation, that appeared already as a trend in the 40 node experiment, is significant and therefore a good capability to differentiate traffic can be attested this strategy combination.

The comparing experiment with a 50 node network revealed, that this strategy combination is able to differentiate the traffic classes in both traffic characteristics. A closer examination of the topology is no longer needed. The trends assumed for the curves shown above were approved.

4.3 SWP (MH) for Best Effort and WSP (ILR) for Delay-Critical Traffic

This section presents another promising strategy combination in our experiments. Now, QoS routing strategies are applied to both traffic classes, namely SWP (MH) for best effort (as before) and WSP for the delay-critical class (as opposed to ECMP). WSP with ILR weight assignment was chosen to encourage the use of short paths while considering the bottleneck bandwith as second routing metric. In combination with SWP, load balancing is expected to be further improved for this experiment. Both strategies reach their routing decision based on the dynamic state of the network, while SWP focuses on using the widest path and WSP on minimizing the hop count.

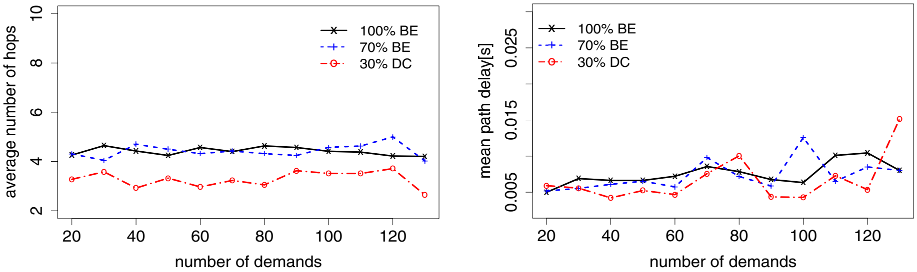

Fig. 6. Traffic characteristics for experiment with SWP (MH) for best-effort class and WSP (ILR) for delay-critical class: average hop count is shown left, mean path delay on the right for partial experiments 1 and 3

The results for the network performance characteristics look alike to the previous 40-node experiment with slightly better values and are not shown here. For figures and further details see [24].

Figure 6 shows the traffic characteristics for this experiment. In comparison to the previous experiment, the curves that display the average hop count are quite similar. On a closer look, they appear to be slightly closer together what may be due to a better load balancing capability of WSP. A quantitative difference of one to one and a half hops can be observed what leads again to a significant differentiation for this characteristic.

With respect to the curves displaying the mean path delays, it can be observed that the dashed and dashed-dotted line are close to each other. Again, the delay-critical class seems to reach somewhat better values thus faster paths are suggested. This strategy combination does not reach a clear differentiation in this traffic characteristic in this scenario but can be identified as suited for an end-to-end delay traffic differentiation. Therefore, the average hop count and assumptions concerning the processing and propagation delays need to be considered together as in the previous experiment evaluation.

The strategy combinations SWP (MH)/WSP (ILR) and SWP (MH)/ECMP (ILR) are quite comparable and their results reveal a good ability to differentiate traffic in IP core networks. Unexpectedly, no major improvements are detected for the application of QoS routing strategies at both traffic classes. For practical reasons ECMP (ILR) should be preferred over WSP (ILR), as the determination of the bottleneck bandwidth is computationally more complex than the determination of shortest paths with static routing weights.

5 Conclusions

The next-generation Internet may tune various different QoS mechanisms to achieve service differentiation between traffic classes. This paper investigates for IP backbone networks with on-demand routing to which extent per-class routing may contribute to this goal. A rather flexible computational framework has been developed in Maple to quantitatively assess various combinations of standard link-metric routing and/or QoS routing strategies.

Results for two traffic classes – delay-critical and best-effort – have shown that a noticeable service differentiation in terms of mean end-to-end delays and mean number of hops may already be achieved with rather fundamental routing schemes. Best results were obtained when SWP (with shortest paths according to MH) is used for the best-effort traffic, while the delay-critical traffic is routed according to WSP (ILR) or even ECMP (ILR). The comparable performance in these two cases is remarkable, since ECMP (ILR) does not take into account the dynamic state of the network. In any case where non-QoS routing is applied, typically for the best-effort class, the ECMP feature is crucial in order to achieve some service differentiation.

In future work, more complex QoS routing approaches will be considered in order to assess the additional benefit of more sophisticated routing procedures – also in the context of more distinct traffic classes with other QoS requirements.

References

1. Toguyeni, A., Korbaa, O.: Quality of service of Internet service provider networks: State of the art and new trends. In: Proc. of Int. Conference on Transparent Optical Networks (2007)
2. Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W.: An architecture for differentiated service. IETF RFC 2475 (December 1998)
3. Braden, R., Clark, D., Shenker, S.: Integrated Services in the Internet Architecture: An Overview. IETF RFC 1633 (June 1994)
4. LeFaucheur, F., et al.: Multi-Protocol Label Switching (MPLS) Support of Differentiated Services. IETF RFC 3270 (May 2002)
5. Colitti, W., Steenhaut, K., Nowe, A.: Multi-layer traffic engineering and DiffServ in the next generation Internet. In: Proc. of 3rd Int. Conference on Communication System Software and Middleware (COMSWARE 2007), pp. 591–598 (2007)

6. Curado, M., Monteiro, E.: A survey of QoS routing algorithms. Trans. Engineering, Computing and Technology (2004)
7. Alsharif, S., Shahsavari, M.M.: Performance study of MPLS and DS techniques to improve QoS routing for critical applications on IP networks. In: Proc. of SPIE - The International Society for Optical Engineering (2008)
8. Bryskin, I., Farrel, A.: GMPLS: Architecture and Applications. Morgan Kaufmann Publishers Inc., San Francisco (2005)
9. Moungla, H., Krief, F.: Service differentiation over GMPLS. In: de Souza, J.N., Dini, P., Lorenz, P. (eds.) ICT 2004. LNCS, vol. 3124, pp. 628–637. Springer, Heidelberg (2004)
10. Guerin, R., Orda, A., Williams, D.: QoS routing mechanisms and OSPF extensions. In: Proc. of Global Internet Miniconference (1997)
11. Medina, A., Lakhina, A., Matta, I., Byers, J.W.: BRITE: An approach to universal topology generation. In: Proc. of 9th Int. Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2001), p. 346. IEEE Computer Society, Los Alamitos (2001)
12. Leduc, G., Abrahamsson, H., Balon, S., Bessler, S., D'Arienzo, M., Delcourt, O., Domingo-Pascual, J., Cerav-Erbas, S., Gojmerac, I., Masip-Bruin, X., Pescapé, A., Quoitin, B., Romano, S.F., Salvatori, E., Skivée, F., Tran, H.T., Uhlig, S., Umit, H.: An open source traffic engineering toolbox. Computer Communications 29, 593–610 (2006)
13. Fortz, B., Thorup, M.: Internet traffic engineering by optimizing OSPF weights. In: Proc. of 19th IEEE INFOCOM, Tel Aviv, Israel, vol. 2, pp. 519–528 (2000)
14. Fortz, B., Rexford, J., Thorup, M.: Traffic engineering with traditional IP routing protocols. IEEE Communications Magazine 40, 118–124 (2002)
15. Wang, Z.: Internet QoS: Architectures and Mechanisms for Quality of Service. Morgan Kaufmann Publishers Inc., San Francisco (2001)
16. Raja, S.V.K., Raj, P.H.: Integrated subset split for balancing network utilization and quality of routing. Proc. of World Academy of Science, Engineering and Technology 20, 96–100 (2007)
17. Crawley, E., Nair, R., Tajagopalan, B., Sandick, H.: A framework for QoS based routing. IETF RFC 2386 (August 1998)
18. Varela, A., Vazao, T., Arroz, G.: Multi-service routing: A routing proposal for the next generation Internet. In: Boavida, F., Plagemann, T., Stiller, B., Westphal, C., Monteiro, E. (eds.) NETWORKING 2006. LNCS, vol. 3976, pp. 990–1001. Springer, Heidelberg (2006)
19. Ash, G.: Traffic Engineering and QoS Optimization of Integrated Voice & Data Networks. Morgan Kaufmann Publishers Inc., San Francisco (2007)
20. Kompella, K., Rekhter, Y.: OSPF Extensions in Support of Generalized Multi-Protocol Label Switching (GMPLS). IETF RFC 4203 (October 2005)
21. Nclakuditi, S., Zhang, Z., David, H.C.D.: On selection of candidate paths for proportional routing. Computer Networks 44, 79–102 (2004)
22. Medhi, D., Ramasamy, K.: Network Routing: Algorithms, Protocols, and Architectures. Morgan Kaufmann Publishers Inc., San Francisco (2007)
23. Wang, Z., Crowcroft, J.: Quality-of-service routing for supporting multimedia applications. IEEE Journal on Selected Areas in Communications 14(7), 1228–1234 (1996)
24. Jensen, U.: Vergleich von Traffic Engineering Strategien fuer IP-Netze mit verschiedenen Verkehrsklassen. Diplomarbeit, Fach Informatik, University of Erlangen, Germany (2008)
25. Maplesoft: Maple 12. Tool for mathematics and modeling, Waterloo, Ontario, Canada (2009), `http://www.maplesoft.com/`