

# Adaptive and Reliable Multipath Provisioning for Media Transfer in SDN-based Overlay Networks

Sahel Sahhaf, Wouter Tavernier, Didier Colle, Mario Pickavet

*Ghent University- IMEC*

---

## Abstract

Traditional routing in the Internet is best-effort which makes it challenging for video streaming since no throughput, jitter, delay or loss rate is guaranteed. As different paths have different characteristics, path differentiation such as multipath routing is a promising technique to be used for meeting QoS requirements of media-intensive applications. Using overlay networks different paths are offered which enable more flexibility in QoS and congestion control while the reliability of the connections is enhanced.

Software Defined Networking (SDN) is known to be a promising solution to the problems of routing as it provides fine-grained control over packet handling. Relying on SDN, we propose an adaptive multipath provisioning scheme ensuring maximal bandwidth and resiliency of media transfer in overlay networks. The scheme is a time slot-based approach which dynamically finds multipaths. It relies on both active probing and traffic prediction.

The experimental results confirm that a more accurate prediction together with more frequent probing lead to fewer number of path re-calculation and also indicate that the proposed scheme enhances the reliability of connections while a more balanced load is achieved in the network compared to the shortest path-based scheme.

*Keywords:* Multipath routing, Overlay network, Software Defined Networking

---

## 1. Introduction

Live streaming is becoming the dominant Internet traffic in the coming years. Industry forecasts that in 2019 up to 90% of the Internet traffic is composed of multimedia streaming (1). The reason is that more enterprises prefer

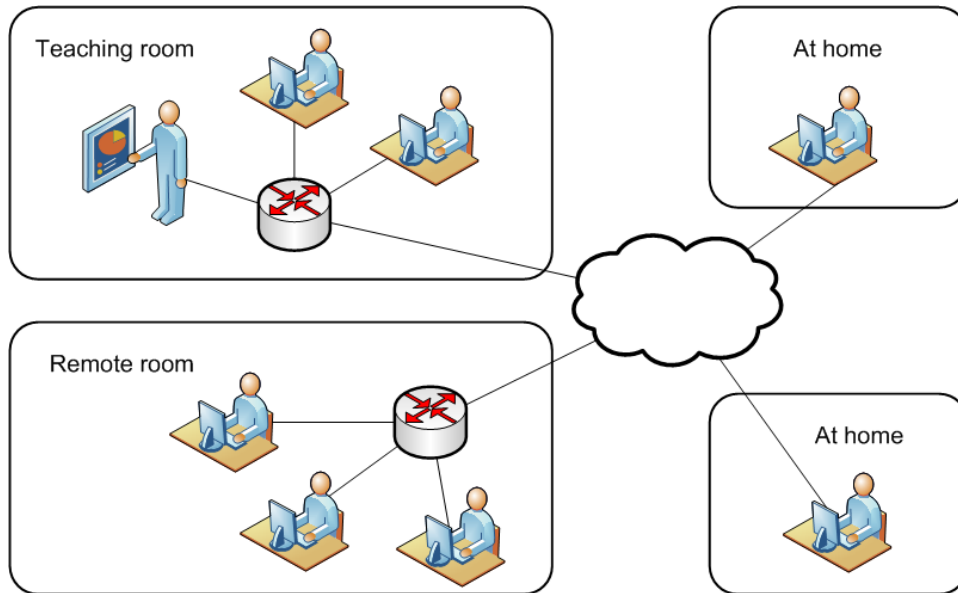


Figure 1: Education use case. A teacher and students exchanging information during class time. They might be in the same room or attend the session remotely via Internet.

to stream on the Internet, e.g., radio and television broadcasts, multimedia conferencing and sport/live events with global audience. Both end-users and broadcasters expectation has increased to a high-quality live viewing experience comparable with high-definition (HD) television broadcast.

One concrete example of live streaming is a distributed educational A/V cooperation which is illustrated in Figure 1. In such a scenario, the teacher is in a class with probably some local students while providing a collaborative experience to remote students which are either joining the session from their home or due to space limitation are located in another remote room. Different contents such as presentations, speeches, student interaction messages and other class materials should be transmitted via local network or through the Internet. However, supporting a high-quality media transfer with different QoS requirements over the Internet is significantly challenging since the traditional routing in the Internet is best-effort and it suffers from impairment such as packet loss, jitter and outages of unknown duration. As a result different QoS requirements cannot be guaranteed.

Overlay networks are considered as promising technology to deliver live streams over the Internet. They form a virtual topology on top of the ex-

isting Internet. The overlay nodes can be located at different Autonomous Systems (ASes) and diverse paths can be found among these overlay nodes. To forward packets through the alternate paths tunneling mechanisms can be exploited. They have been the subject of significant research in the last few years. The main motivations for overlay networks are: i) traditional IP routing is mainly based on shortest path approach and provides the same route independent of the performance in terms of delay, jitter, loss, etc. However, overlay networks enable using different routes depending on the performance metrics such as delay, loss and throughput. ii) Overlay networks are capable of faster failure recovery compared to routing protocols such as BGP. The reason is that in such networks failures are quickly detected through the active probing of the overlay nodes and the traffic can be routed to intermediate overly nodes to bypass the failure and iii) there is no need to change the underlying network (i.e., the Internet). The overlay nodes can be placed in different parts of the network, e.g., at different Data Centers, DCs, end user ISPs or even at IXPs (2). Controlling an overlay network with a central entity enables the service provider to control the path taken from one node (DC) to another over the Internet. Since this central entity is responsible for monitoring the overlay network it can quickly react to the changes/failures in the network by dynamically enforcing new routing instructions to the overlay nodes. Software Defined Networking (SDN) is a promising technology which offers such direct control over packet-processing rules and thus we propose to use a SDN-based overlay network to adaptively find multipaths for live streaming over the Internet.

**Our contribution.** In this work, we propose a SDN-based overlay network architecture for video streaming over the Internet. We then propose a dynamic multipath provisioning approach to be used in such networks. Unlike many existing works which focus on delay and loss rate measurements of the underlying network (3), we consider two metrics of available bandwidth and availability measurements in the process of path selection. The reason is that metrics such as delay or loss rate are not the direct traffic load indicator and they indicate whether a path is already congested. However, available bandwidth indicates the amount of traffic that can still be routed through a path before it is congested. Therefore, relying on such metric we can provide a congestion control mechanism in the network. On the other hand availability of a component indicates the probability that the component is in a functional state at any arbitrary time. This is a significant QoS metric for describing the reliability and we take this into account to enhance network

reliability. Relying on the result of active probing measurements on available bandwidth and the information on links' availability, our proposed algorithm offers an implicit load balancing mechanism, enhances network reliability while trying to reduce the number of path re-calculation to accommodate the time-varying traffic in the Internet efficiently.

The rest of the paper is organized as follows. Section 2 describes the related work. In Section 3, we describe the proposed SDN-based overlay network architecture and suggest some tools to be used for measurement of the considered metrics. Section 4 details the reliability performance parameters which are used in both path selection process and performance evaluation. The problem statement is described in Section 5 and the proposed adaptive multipath provisioning algorithm is detailed in Section 6. The performance evaluations of the proposed approach are reported in Section 7 and finally Section 8 concludes the paper.

## 2. Related Work

In (3) and (4) many of the multipath routing protocols in the current Internet are detailed which can be used for Traffic Engineering (TE) and fulfilling the QoS requirements. The authors review several protocols, from application to link and physical layers. There are several IETF activities in support of inter-domain TE such as (5) and (6). Additionally, several research projects focused on this topic. Some of which are focusing on PCE-based framework. (7) provides a survey on the PCE architecture. The Dragon project, (8), relied on this architecture to implement multi-domain TE paths. There are several proposals to extend BGP to advertise TE information (9), (10) or to have a combination of overlay architecture and BGP extensions (11). The main challenge in such approaches is that they require changing BGP. Furthermore, the exchange of TE information on BGP is not scalable. In a more recent work, relying on inter-AS (G)MPLS tunnels, we proposed a multipath provisioning method for the Internet. The proposed approach relies on a PCE-based architecture. The main limitation of this approach is that it is not designed to be extended to the whole Internet (12).

Overlay networks form a virtual topology on top of the existing networks. Therefore, there is no need to change the underlying network. These networks are highly scalable and flexible which motivate their widespread use. In such networks several paths among overlay nodes can be found which make them

promising solutions to increase resiliency, enable load balancing and meet different QoS requirements.

In some early works (13), the authors noticed that in 30 to 80% of failures in the Internet, there is an alternative path with better characteristics in terms of bandwidth, delay or packet loss. Therefore, they proposed to use an overlay network to take advantage of such paths. Similar to (13), in (14) authors designed a Resilient Overlay Network (RON) to improve the resiliency of the end-to-end connections in the Internet. They relied on active probing of the overlay nodes to react quickly to the failures. In (15), authors tried to improve the scalability of RON by avoiding permanent probing of links. In (16), the network topology was taken into account when the overlay network is constructed. This way they could better control the detour routes. In a more recent work, authors investigated the location of the overlay nodes in the network and suggested to place them at the Internet exchange points (IXPs) (2). They argue that the richness of the connectivity at IXPs can increase the number of alternative paths and thus improve the resiliency.

Live streaming on overlay networks has been investigated in works relying on Akamai architecture (17). Akamai is a large-scale commercial overlay network used for media delivery. Significant effort has been made to make this platform highly scalable as it is composed of more than 147000 servers over the world. A complementary approach is the peer-to-peer (P2P) approach in which end-users machines can self-organize themselves into an overlay tree which is used to distribute media content. Although P2P live streaming is known to be cost-effective, it is not clear whether it can provide the same level of scalability and QoS as provided by dedicated overlay networks. (18) surveys several algorithms and mechanisms considered in P2P overlay networks.

In this work, relying on an overlay architecture, we propose a mechanism to adaptively find multipaths providing load balancing and high availability for live streaming over the Internet. In order to reduce the cost and overhead of path calculation we propose a SDN-based overlay network which is detailed in Section 3. To the best of our knowledge, such adaptive multipath provisioning on SDN-based overlay networks has not been investigated so far.

### **3. Architectural framework: A SDN-based Overlay network**

Relying on Software Defined Networking, our initial goal is to reduce the cost and overhead of path (re)calculation while achieving high resiliency,

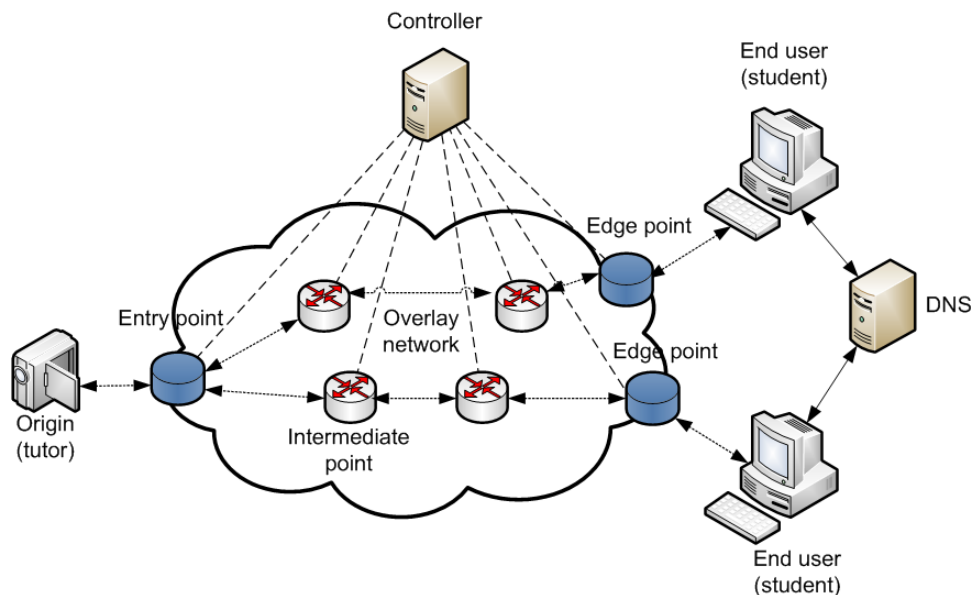


Figure 2: SDN-based overlay network architecture

load balancing and guaranteed QoS of media transfers over the Internet. Additionally, our target is to achieve more flexibility compared to deploying private links or (G)MPLS tunnels over the Internet. Below, we explain the choices made to achieve these goals.

We considered to use the Internet instead of private links or MPLS tunnels to minimize the cost. Overlay networks enable us to deploy our proposed multipath provisioning approach without making any change to the underlying Internet. Using a SDN-based overlay architecture, the Service Providers (SP) can control the traffic themselves and react to different events such as failures.

Figure 2 depicts the main components of the proposed architecture. The overlay network consists of three types of nodes: i) entry points, ii) intermediate points and iii) edge points. These nodes are controlled by a central software element (controller) which is managed by the SP. The entry point works as the source of the A/V streams and forwards each stream to one/several intermediate nodes. The intermediate nodes have essential role in the architecture and can steer traffic in different directions to send the traffic to nodes close to the edge points near the end-users. They behave very similar to SDN switches with the difference that they are not connected directly but

via IP tunnels (e.g., GRE tunnels). As the edge point might receive multiple instances of a stream (due to multipath provisioning) it should combine the packets of different streams, drop the duplicates, recover the lost ones and fix the out-of-order packets to make a single instance of the stream to be sent to the end user.

Since the proposed architecture aims at increasing the resiliency of media transfer across the Internet, a method should be provided to redirect traffic upon failures. This is the role of the central controller to steer traffic away from the failure by enforcing alternative routing paths to the edge/entry points. To this end the controller needs to have a full view of the overlay network. Each overlay node regularly sends probes to the other overlay nodes to check the connectivity and sends the measurement results to the controller. Depending on the result of this active probing the controller can detect failure/impairment and enforce alternative paths. Apart from failures, since we rely on the Internet instead of private links or MPLS tunnels, in order to fulfill the QoS requirements of media intensive applications in terms of bandwidth, delay and jitter, the controller might change the paths for each media transfer several times during its lifetime to meet those QoS requirements. This imposes a large overhead on the controller to reconfigure the flow entries in the overlay nodes. To reduce this overhead, we considered a variation of source routing (19) in which the controller should reconfigure the newly calculated paths only in the entry/edge point of the path.

As mentioned before, in this work we focus on two metrics: i) available bandwidth and ii) availability of the links. There have been several studies proposing bandwidth measurement techniques and tools to be used by the overlay nodes (20). In order to calculate availability, we rely on the availability model of a bidirectional line proposed in (21). Based on this model, fiber optic cable is the dominant component since the cable cuts are frequent and repair times are very long. Accordingly, the availability of a line is dependent on its length. We propose to use tools such as traceroute-geolocation<sup>1</sup> in the overlay nodes to find the geolocation coordinates of each hop along the path in the underlying network. Using these coordinates a rough estimation of the length of the physical links corresponding to each overlay link can be made. Based on these calculations and the models used for calculation of a path availability (detailed in Section 4), the availability of an overlay link is

---

<sup>1</sup><https://nmap.org/nsedoc/scripts/traceroute-geolocation.html>

estimated.

In the proposed overlay network nodes are not physically fixed. The only requirement is that nodes are willing to participate in the control of the overlay network, coordinating their switching behavior in concert with the proposed controller. In case we consider a company with multiple distributed locations with network equipment (offices, data centers, or other), one can safely assume that in each of these locations one can have participating nodes. Alternatively, one could consider renting nodes in datacenters solely for increasing the number of participating nodes, enabling wider flexibility in routing. In the extreme case private partners lend their equipment for routing/switching services by installing a piece of software, similar to SETI or Torrent-servers.

We elaborate the proposed functional architecture considering the education use case described before. Assume that a tutor coordinates a class and there are students who want to join this session remotely (see Figure 2, the end users represent the students and origin is where the tutor is). First each student is mapped to a nearby edge point (the DNS can work based on a load balancing mechanism to map the end users to the edge points). The edge point forwards the requests in the overlay to cross the Internet to reach an entry point close to the origin server. In the entry point, if the path to the requesting edge point exists it is used to forward the traffic (i.e., it is added to the header of each packet arriving from the origin/source routing) otherwise a request is sent to the controller to calculate multipaths for the stream. Note that in the education use case both students and the tutor should be able to send and receive live A/V streams. Therefore, the multipaths in both directions should be provided by the controller. Since we considered source routing these multipaths should only be configured in the edge and entry points. Therefore, in the education use case it is expected that the edge and entry points are capable of similar functions (packet re-ordering, loss recovery, etc.) but this might not be needed for other live streaming use cases. As explained earlier, based on the regular measurements of the overlay nodes, the controller might reconfigure the paths several times in the entry/edge points of the paths.

#### **4. Reliability performance parameter**

In this section, we describe the reliability performance parameters which we use in both path selection process and evaluation of the proposed scheme.



#### 4.1. Component availability

Component's availability indicates the probability that the component is functional at any arbitrary moment. In order to calculate availability MTTR and MTBF of the component should be known. These parameters refer to component's mean time to repair (i.e., the average required time for restoring the component) and mean time between failures (i.e., the average time between 2 consecutive failures of the component) respectively. Formula (1) represent the availability  $A$  based on MTTR and MTBF:

$$A = 1 - \frac{MTTR}{MTBF} \quad (1)$$

#### 4.2. Availability of protected and unprotected path

In order for a path to be functional, all the components along that path should be functional therefore, the availability of a path is calculated based on the availability of components (i.e., nodes and links) along that path which is calculated as follows:

$$A_{unprotected} = \prod_{i \in \text{components}} A_i \quad (2)$$

In Formula (2),  $A_i$  represents the availability of the  $i$ th component along the path. Using recovery techniques such as protection, the availability of a path is enhanced. The reason is that a protected path is available if either the primary or secondary path is available.

$$A_{protected} = A_p + A'_p A_s \quad (3)$$

In Formula (3), the  $p$  and  $s$  are used to indicate the primary and secondary paths respectively.  $A_p$  and  $A'_p$  represent the availability and unavailability of the primary path while  $A_s$  is used to show the availability of the secondary path. Below we show how this formula can be extended to calculate the availability of a system,  $A_{system}$ . By a system, we refer to a scenario when multiple primary and secondary paths exist in the network. In the following formula a single failure is considered.

$$A_{system} = \prod_{i=1}^M A_{p_i} + \sum_{j=1}^M \left( \prod_{k=1; k \neq j}^M A_{p_k} \right) A'_{p_j} A_{s_j} \quad (4)$$

In this formula  $M$  represents the number of primary paths in the system. Based on this formula, either all the  $M$  primary paths should be working or it is possible that one of the primary paths is not functional ( $A'_{p_j}$ ) but all the other primary ones should be available together with the secondary path corresponding to the primary path which is unavailable ( $A_{s_j}$ ).

## 5. Problem statement

In this section we formally present the problem of multipath provisioning on the overlay network. This model is used by the controller to regularly find multipaths to transfer media-intensive applications over the Internet while meeting their QoS requirements. We present the overlay network as a directed graph  $G(N, E)$  with nodes  $N$  and edges  $E$ . Relying on the active probing, different metrics such as available bandwidth, delay, jitter and loss can be measured between the overlay nodes. We consider two metrics but the model can simply be extended to include other metrics as well. The first one is the available bandwidth. Each edge in the overlay network has certain capacity in terms of available bandwidth  $B_e$ . The other metric that is considered for each edge is its availability  $A_e$ . Based on these metrics each edge is assigned a cost  $cost_e$ :

$$cost_e = -\ln(A_e) + \alpha \cdot B_e^{-\kappa}, \kappa \in \mathfrak{R}^+ \quad (5)$$

Since in our proposed multipath provisioning scheme, we target minimizing the cost of the paths, considering such a cost for each edge leads to selection of paths with higher availability (enhancing connection reliability) while a more balanced load is achieved. Giving higher costs to the links with less bandwidth enables selection of less loaded links and thus an implicit congestion control is provided. In Formula (5), we considered  $\log$  of the availability. The reason is that in the calculation of a connection's availability, the product of the components availability along the connection is considered (see Formula (2)). The two parameters  $\alpha$  and  $\kappa$  are defined to tune the impact of the factors in the cost function.

The problem is assumed to be an online problem which means that the media transfer requests arrive over time. Each request is defined by a tuple  $r = (s^r, d^r, b^r)$ . In this tuple  $s^r$  and  $d^r$  are the source and destination of the request respectively.  $b^r$  represents the demand of the request in terms of bandwidth. We try to find paths for each request optimally upon its

arrival. Optimality refers to using edges with higher availability and using less popular edges by assigning higher costs to the edges with less available bandwidth.

All the notations used to define the model are reported in Table 1. Using these notations, we propose a heuristic solution to be used by the controller for provisioning multipaths in the overlay architecture.

Variable	Description
$N$	Set of network nodes.
$E$	Set of network edges.
$B_e$	Available Bandwidth in edge $e$ .
$cost_e$	The cost of edge $e$ .
$A_e$	The availability of edge $e$ .
$R$	Set of concurrent requests.
$s^r$	Source node of request $r$ .
$d^r$	Destination node of request $r$ .
$b^r$	Bandwidth demand of request $r$ .

Table 1: Symbols and notations in the formal models

## 6. Multipath provisioning in overlay network

Based on the explained architecture, the controller is the entity which is responsible for (re)calculating the paths for each media transfer request. In this section, we first explain the traffic characteristics in the Internet and then detail the proposed scheme to calculate multipaths in the overlay network.

### 6.1. Time-varying traffic in the Internet

Different investigations indicate that the real-world data traffic may fluctuate randomly in a short period of time which can be within one hour. However, it is also observed that such a change follows a similar pattern in a longer period over time (22), (23). The Internet traffic follows a similar pattern within 24 hours due to the fact that the network users follow a same routine in their daily usage (24). This periodic pattern is called time-varying traffic (25) which is depicted in Figure 3 and as we see the pattern is repeated after a period. In order to deal with time-varying traffic in the Internet, we propose a time slot-based approach in the overlay network, which dynamically re-calculates paths to adapt to the changes in the underlying traffic.

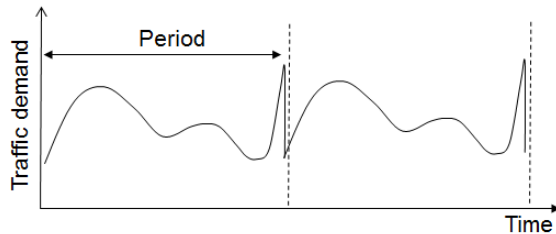


Figure 3: Example of time varying-traffic

### 6.2. Time slot-based Availability and Bandwidth-aware multipath Routing (TABR)

Since the problem of finding optimal survivable multipaths with least cost is NP-hard (26), it cannot always be solved in a reasonable time. Therefore, we propose a heuristic approach which is referred to as TABR.

In this work, we search for two sets of paths for each request, namely primary and secondary. The paths within each set can share links among themselves however the two sets should be disjoint to enhance reliability of the connections.

We consider that the time is divided into fixed length time slots (e.g.,  $n$  second time slots) and the proposed algorithm, iteratively processes these time slots and may (re)calculate paths for each request depending on the measurements results gathered from the active probing of the overlay nodes.

In this algorithm, the requests are sequentially processed. Figure 4 depicts different blocks in the TABR algorithm. New requests are given to the TABR through an API. Before processing the new requests, it is first checked if the previously accepted requests are finished so that their assigned resources can be released. Then in each time slot the resources are updated based on the measurement results of the overlay nodes and the requests which their (QoS) demands are not fulfilled are detected and then a network allocation algorithm (ABMR) is called for each request (new or previously admitted which requires change). If this algorithm is successful in allocating bandwidth to the request, it is admitted and the resources for that time slot is updated otherwise, the previous resource allocation does not change.

The TABR algorithm is shown in Algorithm 1. This algorithm runs over time and given all requests (new and/or previously admitted) in the system, it checks whether any of the previously admitted ones is finished so that its resources can be released. Then it calls the TimeSlot algorithm to process

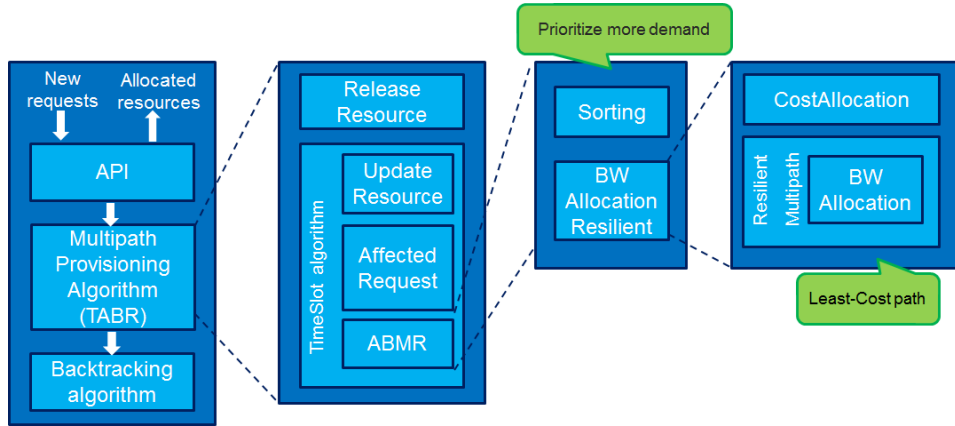


Figure 4: Components of TABR algorithm

the new requests which is depicted in Algorithm 2. Before processing the requests in the system, this algorithm checks if it is the beginning of a new time slot. In this case, it first checks the measurements received from different overlay nodes to update the status of the overlay network in terms of different metrics such as available bandwidth.

```

Data: requests, overlay network
for  $r \in requests$  do
  if  $r$  is finished then
    ReleaseResource( $r$ );
    Remove  $r$  from requests;
  end
end
TimeSlot(requests, network);
Algorithm 1: Time slot-based Availability and Bandwidth-aware Routing (TABR)

```

In our proposed scheme, we only consider available bandwidth between overlay nodes however this can be extended to different metrics which can be measured through active probing (delay, loss, jitter). The UpdateResource module is responsible for determining the available bandwidth gathered from the active probing of the overlay nodes which is stored in actualBW. As mentioned before, different studies show that the Internet traffic follows a

periodic pattern and thus there are different techniques to predict the traffic for both short and long periods. We propose to use predictions such as (27) in our scheme in order to reduce the requirement for changing the paths of each request regularly over its life time. As we see in the algorithm, in addition to the actualBW, we take minPBW which is the minimum predicted available bandwidth of each link in the overlay network until the next time slot (assuming  $i$  is the length of the time slot). Using this prediction, we assume that the available bandwidth in the current time slot,  $t$ , which can be used for multipath provisioning of the requests arriving in  $t$  is equal to the minimum of actualBW and minPBW (TimeSlotBW[t]).

The algorithm then checks whether any of the already admitted requests are affected. This is identified from the results of the measurements performed by the overlay nodes. The affectedRequest module is responsible for checking the requests and if the performance of any request is below a certain threshold it is added to the affectedR list to be re-configured. The ABMR algorithm is then called first for these requests as they have higher priority compared to the newly arrived requests. Once these requests are processed the algorithm calls ABMR to process the new requests.

We believe that taking a traffic prediction model into account leads to better resource allocation and decreases the chance of multipath recalculation. Indeed this is dependent on the accuracy of the prediction which is evaluated in Section 7.

Note that in this algorithm, it is assumed that the overlay nodes send their measurements regularly at the beginning of each time slot.

```

Data: requests, overlay network
for  $t \in time$  do
  if  $t$  is beginning of time slot then
    actualBW  $\leftarrow$  UpdateResource(active probing);
    minPBW  $\leftarrow$  min(predicted available bandwidth until  $t+i$ );
    TimeSlotBW[t]  $\leftarrow$  min(actualBW, minPBW);
    affectedR  $\leftarrow$  affectedRequest(requests);
    ABMR(affectedR);
  end
  ABMR(requests);
end

```

**Algorithm 2:** TimeSlot

In the Availability and Bandwidth-aware Multipath Routing (ABMR) algorithm, first the (concurrent) requests are sorted based on their bandwidth demand in a descending order. For each request, the network resource usage is maintained as depicted in Algorithm 3 which is used for backtracking in case of failure.

```

Data: requests, overlay network
sortedList ← sortBandwidth(all concurrent requests);
for request ∈ sortedList do
  | currentState ← save the current network state;
  | feasible ← BWallocationResilient(request);
  | if feasible then
  | | update the network;
  | else
  | | set current network state to currentState;
  | end
end

```

**Algorithm 3:** Availability and Bandwidth-aware Multipath Routing (ABMR)

Each request is then given to the BWallocationResilient algorithm which finds the close to optimal paths for each request. If a feasible set of paths for both primary and secondary paths is found, the network is updated based on the new allocation, otherwise the algorithm backtracks to the previous state of the network.

```

Data: a request
costAllocation(edges);
primaryReservation ← BWallocation(req, demand, graph);
graphReduced ← remove the links in primaryReservation from the
  network graph;
backupReservation ← BWallocation(req, demand, graphReduced);
if primaryReservation && backupReservation then
  | return primaryReservation, backupReservation;
else
  | return false;
end

```

**Algorithm 4:** BWallocationResilient

The BWallocationResilient is responsible for finding multipaths as primary and secondary which is shown in Algorithm 4. In this algorithm first the costAllocation module assigns a cost according to Formula (5) to each edge of the overlay network (based on the edge availability and available bandwidth calculated for that time slot, i.e., TimeSlotBW[t]). Then the BWallocation module is called for each request for the first time to find a set of multipath as primary set. This set fulfills the request’s demand in terms of bandwidth. Then the links within this set are removed from the network and the BWallocation is called for the second time on the reduced topology to find second set of multipaths as secondary/backup set. Note that if resiliency against a single link failure is required, the amount of backup bandwidth can be reduced. To this end, it is sufficient to only reserve bandwidth equal to the maximum bandwidth allocated on the edges of the primary paths. As such, if any link in the primary paths breaks there is enough backup bandwidth for the recovery of the affected paths. However, in order to enhance the reliability, we can find two sets of disjoint paths with bandwidth equal to the request’s demand. This way we can duplicate the streams on two sets of paths so that the edge point can recover the streams in case of packet loss.

If a feasible set of primary and secondary paths are found the request is admitted and the network is updated according to the new allocations otherwise the request is rejected.

Note that the algorithm does not prefer multipaths over single path in each set. By preferring the paths with more available bandwidth (considered in the edge cost), the algorithm first tries to select a single path for each set and only if there is not enough capacity in a single path to fulfill the requests demand, it iterates to find multipaths. The reason is that in video streaming, we prefer to reduce the traffic splitting so that the packet re-ordering (out of order packet problem) is less challenging in the edge points.

Now we explain the main block of the algorithm which is called BWallocation and is shown in Algorithm 5. This algorithm iteratively finds the shortest augmenting paths in the residual network and augments flows along such paths until the demand of the request is fulfilled. The shortest paths are found based on a modified version of the Bellman-Ford algorithm which considers the calculated edge cost as their weight.



```

Data: a request, demand, graph
path  $\leftarrow$  LeastCostPath(graph);
while path do
    minBW  $\leftarrow$  min Bandwidth(path);
    flow  $\leftarrow$  flow + minBW;
    if flow  $\geq$  demand(req) then
        minBW  $\leftarrow$  minBW - (flow - demand(req));
        reservation(req, flow);
        return reservation;
    else
        augment flow along path with minBW
        update the residual graph;
    end
    path  $\leftarrow$  LeastCostPath(graph);
end
return false;

```

**Algorithm 5:** BWallocation

## 7. Performance evaluation

We first describe our simulation environment and explain the performance metrics which are used in the evaluation of the schemes. Then we report the results of the evaluation. As the proposed scheme focuses on enhancing network reliability and providing load balancing the performed evaluations concentrate on quantifying the added value achieved in terms of these metrics. We compare the proposed TABR with another multipath routing based on shortest path algorithm. Additionally we evaluate the impact of different request arrival rates and the size of the time slots on different performance metrics.

### 7.1. Simulation environment

Our simulation environment is based on Python code in which we used Networkx library for graph-based implementations. We implemented our proposed TABR algorithm and a multipath routing approach based on shortest path algorithm for comparison. We considered an overlay network composed of 10 nodes which is fully connected. The available bandwidth of the links are numbers uniformly distributed in the range of [100, 200] and links availability are numbers in the range [0.9,0.99999].

The requests arrive over time in a Poisson process with average rates between 2 and 10 requests per 100 time units. It is assumed that each request has a lifetime exponentially distributed with an average of  $\mu = 1000$  time units. The source and destination for each request is selected randomly and its bandwidth demand is number uniformly distributed in the range of [10, 50]. For the calculation of links' cost in TABR algorithm, it is assumed that  $\alpha = \kappa = 1$ .

In addition to the overlay traffic, we simulate some non-overlay traffic to account for the traffic in the underlying network. Obtaining a realistic model for Internet traffic is challenging as traffic data is rather scarce. Based on the investigation in (28), the traffic distribution resides somewhere between a Pareto and a log-normal distribution. In this experiments, we generate traffic with Pareto distribution between every two overlay nodes indicating the traffic in the underlying path between that two nodes. It is assumed that this traffic is at most equal to the available bandwidth on the corresponding link and thus no congestion can occur.

Each simulation run covers a 12 hour period and a hardware with Intel Xeon quad-core CPU at 2.40 GHz with 12 GB RAM is used for running the simulations. The simulations are iterated 20 times and the average result over all the iterations is reported.

### 7.2. Performance metrics

For the evaluation of the proposed scheme the following performance metrics are considered.

**Availability.** This metric was explained in Section 4 and it is the probability that a component is functional at any arbitrary time.

**Load balancing.** This metric is defined to measure the load balancing performance of a routing mechanism objectively.

**Acceptance ratio/rate.** It measures the ratio between the number of requests for which multipaths could be found upon their arrival and the total number of requests.

**Re-configuration ratio.** This metric indicates the ratio between the number of requests for which multipaths should be re-calculated and the total number of requests.

### 7.3. Comparison with shortest path

In the first set of experiments, we evaluate the proposed TABR in terms of the explained performance metrics and compare the results with the mul-

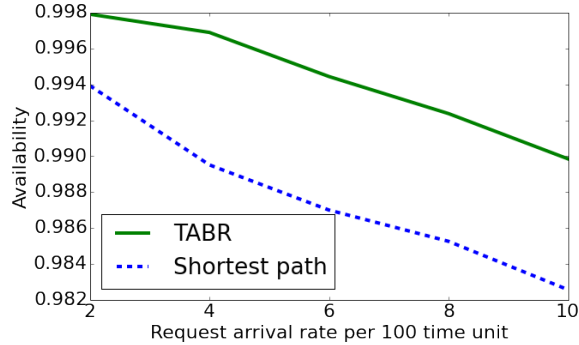


Figure 5: Comparison of connection availability.

tipath routing approach based on the shortest path. In the following experiments, the time slot of 5s is considered unless stated otherwise.

### 7.3.1. Availability analysis

In order to perform an availability analysis of the proposed scheme, we calculated the availability of the paths based on Formula (4). Figure 5 depicts the result of this evaluation. In this figure, the average connection availability which is achieved for the requests with different arrival rates is depicted. The results are compared with the shortest path-based approach. As we see, compared to the shortest path TABR achieves a higher connection availability since links availability is taken into account at the time of path selection. As a result a more reliable network is achieved. The results indicate that for higher request arrival rates, lower availability is achieved. This is due to the fact that higher arrival rate results in the presence of more flows at the same time in the network. Accordingly, there are fewer possibilities for path selection and therefore, for many requests a path with low availability is selected.

### 7.3.2. Load balancing analysis

In order to evaluate the load balancing properties of a routing scheme we need to define a load balancing metric. The existing metrics in the literature are originally designed to be used for evaluating the load balancing of nodes. However, they can simply be extended to the link load balancing metric. We consider a metric called  $\beta$ -ratio defined in (29). It can be used for evaluating the load balancing properties of nodes or links in the network and is defined as:

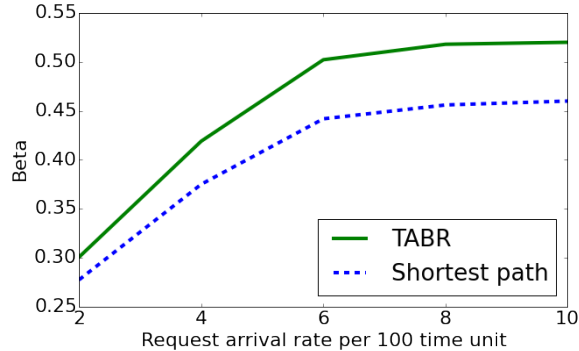


Figure 6: Comparison of  $\beta$ -ratio.

$$\beta = \frac{(\sum_{x \in X} f_x)^2}{|X| \sum_{x \in X} f_x^2} \quad (6)$$

In this metric,  $X$  is the set of vertices  $V$  or edges  $E$  of the network  $G = (V, E)$  and  $f_x$  indicates the number of paths going through vertex or edge  $x$ . Note that this ratio ranges between  $\frac{1}{X} \approx 0$  (unbalanced) to 1 (balanced).

Since we take the available bandwidth of the links into account in path selection process, an implicit load balancing mechanism is enabled in the network (by preferring the links with more available bandwidth). The results of  $\beta$ -ratio evaluation indicate a better load balancing property in the proposed scheme compared to the shortest path-based approach which is illustrated in Figure 6. As we see in this figure, the beta increases for higher arrival rates. This can be explained by the fact that when there are fewer flows in the network (i.e., for lower arrival rates) fewer links are used compared to the scenario when more flows are in the network at the same time. Therefore, the higher number of unused links compared to the used ones by the existing flows leads to a more unbalanced load (lower  $\beta$ ).

### 7.3.3. Acceptance ratio

The next experiment evaluates the acceptance ratio of the proposed scheme compared to the shortest path-based approach for different request arrival rates. The results are depicted in Figure 7 which indicates the lower acceptance rate for higher arrival rates which is expected since fewer resources are available in the network in case of the latter.

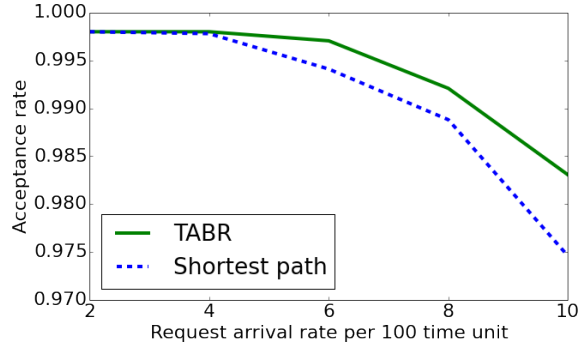


Figure 7: Comparison of acceptance ratio/rate.

#### 7.4. Impact of request arrival rate

In the next experiments we evaluate the impact of request arrival rate on several metrics. In these evaluations, we consider different percentage of error in the predicted traffic to identify the impact of prediction accuracy on the performance of the scheme.

First experiment evaluates the acceptance ratio of the proposed scheme vs different arrival rates while different percentage of error is considered. Figure 8 illustrates the result. Similar to previous experiment, fewer resources in the network (as a result of higher arrival rate) leads to lower acceptance rate of the requests. However, a more accurate prediction of the future traffic improves the performance.

The next experiment evaluates the re-configuration ratio in the proposed scheme. Figure 9 depicts this ratio for different request arrival rates. As we see in this figure, the higher the error in prediction, the higher the number of requests which should be re-configured. Additionally we see that the increase in the request arrival rate leads to higher re-configuration ratio.

In addition to the re-configuration ratio, we evaluate the number of times that the paths of a request should be re-configured during its life time and report the average number over all the accepted requests in Figure 10. We observe an exponential trend in the increase of this number vs. the arrival rate of the requests.

Note that, in practice, it may happen that for some requests which were already accepted, no path can be found to fulfill their demand in the new time slots (after updating measurement result performed by the overlay nodes). The next experiment illustrates the percent of the initially accepted requests

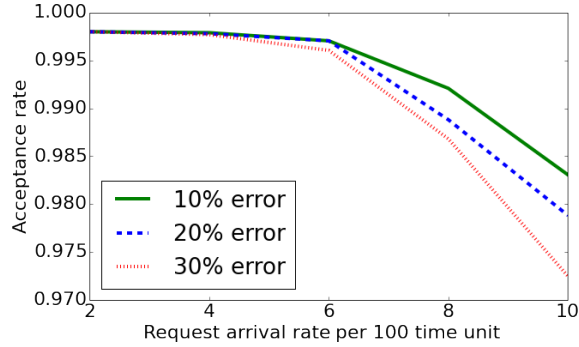


Figure 8: Acceptance ratio/rate for different request arrival rates with different percentage of error.

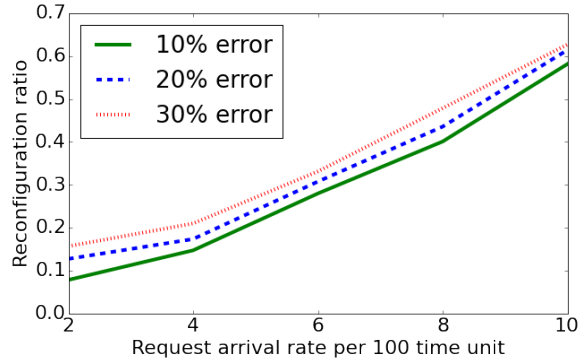


Figure 9: Re-configuration ratio for different request arrival rates with different percentage of error.

which could not be re-configured in the coming time slots. We indicate this in Figure 11 with the percent of rejection. Indeed for low arrival rates there are enough resources to fulfill the demand of all requests in the system however, this percentage increases with the increase of the requests arrival rates.

### 7.5. Impact of time slot size

In the proposed scheme, a longer time slot decreases the overhead caused by regular measurements (active probing) and reporting to the controller, as a result the staleness of the overlay routing information is increased. In the next experiment, we evaluate the impact of different size for time slot on the number of path re-configuration and re-configuration ratio. In this experiment, the fixed arrival rate of 6 request per 100 time unit is considered.

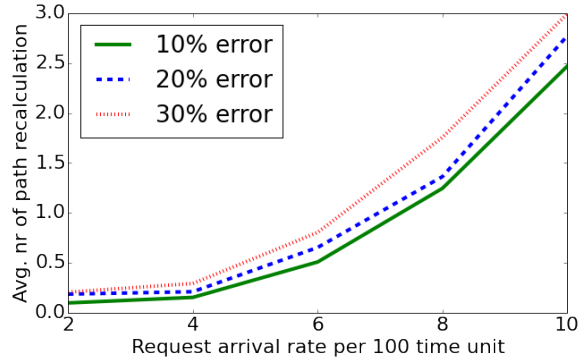


Figure 10: Number of path re-configuration for different request arrival rates with different percentage of error.

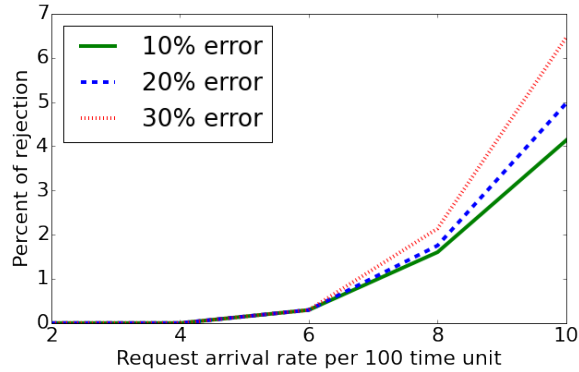


Figure 11: Percent of rejection for different request arrival rates with different percentage of error.

Figure 12 illustrates the reconfiguration ratio vs. different time slots while different percentage of error in traffic prediction is considered. The average number of path re-calculation is depicted in Figure 13. Note that while the overhead of regular measurements decreases, fewer requests are given their full demand. This is visible by having 30% path re-configuration for 5s time slot vs. 20% in case of 25s time slot (Figure 13).

Experimental results in this section confirmed the advantage of the TABR algorithm compared to the shortest path-based approach in terms of load balancing, connection availability/reliability and acceptance rate. Using an accurate traffic prediction model, the network is more efficiently used (indi-

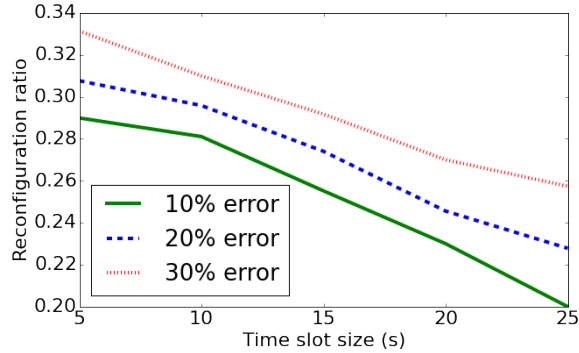


Figure 12: Re-configuration ratio for different time slot size with different percentage of error.

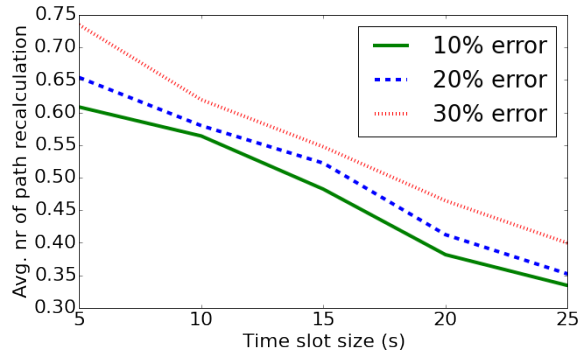


Figure 13: Number of path re-configuration for different time slot size with different percentage of error.

cated with a higher acceptance rate) and the number of path re-configuration for the requests is reduced. The results indicated a trade-off between the overhead of the probing and the fulfillment of the QoS requirements of the requests. This means that the more frequent probing leads to better/faster detection of impairments which in turn leads to faster adaptation to network states.

## 8. Conclusion

In this paper, we investigated the problem of finding multipaths adaptively to increase bandwidth and connection reliability for media transfer over the Internet. To reduce cost and avoid changes to the underlying network,



we proposed a SDN-based overlay network architecture. This architecture provides a central control enabling the provider to control the paths taken from one node to another. This way the overlay network can quickly react to changes/failures in the network. We then proposed an adaptive time slot-based multipath provisioning approach relying on the active probing measurements of available bandwidth and the information on links' availability. The proposed approach offers an implicit load balancing mechanism, enhances network reliability while reducing the number of path re-calculation required to accommodate the time-varying traffic in the Internet. The simulation results indicated that the proposed scheme have better performance in terms of acceptance rate, connection availability and load balancing in comparison with the shortest path-based scheme. Additionally, the impact of traffic prediction on the performance of the scheme in terms of acceptance rate and number of path re-calculation was evaluated. The results confirmed that a more accurate traffic prediction combined with more frequent probing can lead to a better path selection which reduces the number of path-recalculation for media transfers during their life time. Future works include implementation of a prototype of the proposed scheme in a SDN framework.

## Acknowledgment

The research described in this paper is partially funded by the iMinds EMD project and by the UGent BOF/GOA project 'Autonomic Networked Multimedia Systems'.

## References

- [1] C. V. N. Index, Forecast and methodology, 2014-2019 white paper, Tech. rep., Technical Report, Cisco (2015).
- [2] A. Fressancourt, C. Pelsser, M. Gagnaire, Kumori: Steering cloud traffic at ixps to improve resiliency, in: Design of Reliable Communication Networks, 2016. (DRCN 2016). Proceedings., IEEE, 2016.
- [3] S. K. Singh, T. Das, A. Jukan, A survey on internet multipath routing and provisioning, Communications Surveys & Tutorials, IEEE 17 (4) (2015) 2157–2175.

- [4] S. Qazi, T. Moors, Finding alternate paths in the internet: A survey of techniques for end-to-end path discovery, arXiv preprint arXiv:1310.8125.
- [5] A. Farrel, A. Ayyangar, J. Vasseur, Inter-domain mpls and gmpls traffic engineering–resource reservation protocol-traffic engineering (rsvp-te) extensions, Internet Engineering Task Force, RFC 5151.
- [6] R. Zhang, Mpls inter-autonomous system (as) traffic engineering (te) requirements.
- [7] F. Paolucci, F. Cugini, A. Giorgetti, N. Sambo, P. Castoldi, A survey on the path computation element (pce) architecture, *Communications Surveys & Tutorials*, IEEE 15 (4) (2013) 1819–1841.
- [8] T. Lehman, J. Sobieski, B. Jabbari, Dragon: a framework for service provisioning in heterogeneous grid networks, *Communications Magazine*, IEEE 44 (3) (2006) 84–90.
- [9] G. Stea, E. Mingozzi, The euqos system: A solution for qos routing in heterogeneous networks, *IEEE Communications Magazine* (2007) 3.
- [10] D. Griffin, J. Spencer, J. Griem, M. Boucadair, P. Morand, M. Howarth, N. Wang, G. Pavlou, A. H. Asgari, P. Georgatsos, Interdomain routing through qos-class planes [quality-of-service-based routing algorithms for heterogeneous networks], *Communications Magazine*, IEEE 45 (2) (2007) 88–95.
- [11] W. Xu, J. Rexford, MIRO: multi-path interdomain routing, Vol. 36, ACM, 2006.
- [12] S. Sahhaf, W. Tavernier, D. Colle, M. Pickavet, Resilient availability and bandwidth-aware multipath provisioning for media transfer over the internet, in: *Resilient Networks Design and Modeling (RNDM)*, 2016 8th International Workshop on, IEEE, 2016, pp. 134–141.
- [13] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, et al., Detour: Informed internet routing and transport, *Micro*, IEEE 19 (1) (1999) 50–59.

- [14] D. Andersen, H. Balakrishnan, F. Kaashoek, R. Morris, Resilient overlay networks, *ACM SIGCOMM Computer Communication Review* 32 (1) (2002) 66–66.
- [15] P. K. Gummadi, H. V. Madhyastha, S. D. Gribble, H. M. Levy, D. Wetherall, et al., Improving the reliability of internet paths with one-hop source routing., in: *OSDI*, Vol. 4, 2004, pp. 13–13.
- [16] J. Han, D. Watson, F. Jahanian, Enhancing end-to-end availability and performance via topology-aware overlay networks, *Computer Networks* 52 (16) (2008) 3029–3046.
- [17] K. Andreev, B. M. Maggs, A. Meyerson, J. Saks, R. K. Sitaraman, Algorithms for constructing overlay networks for live streaming, *arXiv preprint arXiv:1109.4114*.
- [18] A. Malatras, State-of-the-art survey on p2p overlay networks in pervasive computing environments, *Journal of Network and Computer Applications* 55 (2015) 1–23.
- [19] M. Soliman, B. Nandy, I. Lambadaris, P. Ashwood-Smith, Source routed forwarding with software defined control, considerations and implications, in: *Proceedings of the 2012 ACM conference on CoNEXT student workshop*, ACM, 2012, pp. 43–44.
- [20] A. Botta, A. Davy, B. Meskill, G. Aceto, Active techniques for available bandwidth estimation: comparison and application, in: *Data Traffic Monitoring and Analysis*, Springer, 2013, pp. 28–43.
- [21] J.-P. Vasseur, M. Pickavet, P. Demeester, *Network recovery: Protection and Restoration of Optical, SONET-SDH, IP, and MPLS*, Elsevier, 2004.
- [22] M. F. Zhani, H. Elbiaze, F. Kamoun, Analysis and prediction of real network traffic, *Journal of networks* 4 (9) (2009) 855–865.
- [23] Y. Chen, S. Jain, V. K. Adhikari, Z.-L. Zhang, K. Xu, A first look at inter-data center traffic characteristics via yahoo! datasets, in: *INFOCOM, 2011 Proceedings IEEE*, IEEE, 2011, pp. 1620–1628.
- [24] S. Sundaresan, W. De Donato, N. Feamster, R. Teixeira, S. Crawford, A. Pescapè, Broadband internet performance: a view from the gateway,

- in: ACM SIGCOMM computer communication review, Vol. 41, ACM, 2011, pp. 134–145.
- [25] Z. Ye, X. Cao, X. Gao, C. Qiao, A predictive and incremental grooming scheme for time-varying traffic in wdm networks, in: INFOCOM, 2013 Proceedings IEEE, IEEE, 2013, pp. 395–399.
- [26] S. Rai, O. Deshpande, C. Ou, C. U. Martel, B. Mukherjee, Reliable multipath provisioning for high-capacity backbone mesh networks, IEEE/ACM Transactions on networking 15 (4) (2007) 803–812.
- [27] H. Tong, C. Li, J. He, Y. Chen, Internet traffic prediction by w-boost: classification and regression, in: International Symposium on Neural Networks, Springer, 2005, pp. 397–402.
- [28] J. Mikians, A. Dhamdhere, C. Dovrolis, P. Barlet-Ros, J. Solé-Pareta, Towards a statistical characterization of the interdomain traffic matrix, in: NETWORKING 2012, Springer, 2012, pp. 111–123.
- [29] H. Velayos, V. Aleo, G. Karlsson, Load balancing in overlapping wireless lan cells, in: Communications, 2004 IEEE International Conference on, Vol. 7, IEEE, 2004, pp. 3833–3836.