Q-POINT: QoE-Driven Path Optimization Model for Multimedia Services

Ognjen Dobrijevic[1]*, Andreas J. Kassler[2], Lea Skorin-Kapov[1], and Maja Matijasevic[1]

[1] University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia
{ognjen.dobrijevic, lea-skorin.kapov, maja.matijasevic}@fer.hr
[2] Karlstad University, Department of Mathematics and Computer Science
Universitetsgatan 2, 65188 Karlstad, Sweden
andreas.kassler@kau.se

Abstract. When delivering multimedia services over Internet, different media types are impacted by resource limitations in a different way. While an interactive audio service calls for low-latency communication, video streams should be routed over network paths with sufficient capacity. However, in current networks flows towards the same destination follow the same path, which may lead to a suboptimal resource utilization that effectively penalizes end-users' quality of experience (QoE). This paper proposes Q-POINT, a QoE-driven path optimization model to fairly maximize aggregated end-user QoE for competing clients' service flows by calculating the best path for each flow, subject to resource constraints. We formulate the problem as a mixed integer linear program integrating QoE models for audio, video and data transfer. Such an approach can be leveraged within the software-defined networking paradigm, which provides a control plane to orchestrate path set-up. We evaluate our model and illustrate its benefits over shortest path selection.

Keywords: Multimedia services, quality of experience, software-defined networking, network-wide optimization, mixed integer linear program

1 Introduction

The Internet is transforming from a data-centric network towards a network that delivers diverse services, accessed by fixed and mobile users. In addition, novel services such as cloud computing or multi-player online gaming lead to a significant increase in traffic demands, which might result in network congestion, thus calling for new resource management mechanisms. When delivering multimedia services over heterogeneous networks, the impact of resource limitations manifesting themselves as, e.g., packet loss or delay depends on the service type and the end-users' quality expectations. For example, dropping an I-frame for a video session might lead to more adverse effects then dropping a few TCP packets for

* Corresponding author

a Cloud-delivered system upgrade. Researchers are looking into new ways that enable flexible, yet efficient optimization of multimedia delivery under resource constraints, while considering user quality, or quality of experience (QoE) [1].

Let us consider a network operator and its network. The operator's goal is to dynamically allocate network resources across all users and services in such a way that the total QoE is maximized over all ongoing sessions, while also considering given resources constraints. However, such an optimization is difficult to achieve because different services may have different resource demands, be impacted differently by resource limitations, and, finally, end-users may have different preferences. This requires metrics that quantify expected QoE of an end-user with regards to a given service and a specific network configuration, and functions that map network resource limitations to service metrics. Once these metrics and mapping functions are in place, the network operator can perform an optimization that guides the resource allocation and leads to, e.g., network path selection for given flows and queue configuration. Such a QoE-based optimization may consider user-, network-, and service-related constraints, but must also regard multiple sessions, service flows, and the whole network domain [2]. A preliminary approach for a multi-user domain-wide optimization has been presented in [3], but was treating the network as a "black box". This has the disadvantage that no control over the resource allocation could be exercised.

In current networks, all flows for a source-destination pair typically follow the same path, which might be a suboptimal decision. Rather, a flow should be routed over a path which has the least impact on QoE degradation for given resource constraints. For example, an audio flow should be delivered over a path that offers low latency. This calls for a mechanism that calculates the "best available path", in terms of the impact on overall QoE, for each service flow and enables per-flow routing conformed to given QoE constraints. Software-defined networking (SDN) [4] proposes an efficient means to decouple data forwarding from the control in network devices. Using, e.g., the OpenFlow protocol, routers can be configured by a centralized control ("SDN controller") to forward flows along certain paths and treat the flows according to quality of service (QoS) rules. As an outcome, SDN-based routing is beneficial for QoE-based optimization [5].

In this paper, we tackle the problem of finding the best path for each media flow by developing Q-POINT, a QoE-driven path optimization model. Our goal is to maximize the aggregated user-expected QoE value over all users and service flows in a network domain, subject to resource constraints and network topology. We use different QoE functions that map resource limitations (i.e., QoS parameters) to the QoE values in terms of mean opinion score (MOS). We formulate the problem as a mixed integer linear program and use linearization techniques to cope with the non-linearity of, e.g., buffering latency. A preliminary evaluation for different network topologies and different number of flows shows that our approach increases the overall QoE over shortest path selection.

The rest of the paper is organized as follows. We review related work in the areas of QoE-based routing and SDN in Section 2. Section 3 presents the proposed path optimization model, along with its mathematical formulation, while Section 4 gives a brief overview of our model implementation. Q-POINT is evaluated in Section 5, followed by the conclusion and future work plans.

2 Related Work

2.1 Path Assignment Based on QoS/QoE Metrics

QoS-based routing has been an active research area going back over the past two decades [6, 7], focused on solving multi-constrained path and constrained shortest path problems. In recent approaches, Kumar *et al.* [8] present multi-objective optimization algorithms aimed at finding optimal routes for service flows belonging to different QoS classes, which is based on the importance of QoS parameters for a specific flow. Given that QoS-based routing as a multi-constrained path problem is known to be NP-complete [9], the authors propose an evolutionary algorithm that considers prioritized QoS requirements. Further, Lu *et al.* propose a genetic algorithm for solving multi-constrained routing problem with QoS guarantees, shown to be efficient in dynamic environments [10].

While QoS-based solutions consider media flows in terms of different QoS parameters and classes, QoE-driven approaches generally incorporate application-level knowledge (e.g., application state or codecs used) which provides more accurate insight to impacts on user quality. Amram *et al.* [11] present network-level mechanisms that support optimization of video transfer in cellular networks. Their goal is to maximize QoE for video flows by calculating needed transmission rate and identifying the optimal network path from video sources, and they equalize QoE among the flows that are delivered through a congested network part. A QoE optimization approach based on overlay networks that routes traffic around link failures and congestion is proposed by De Vleeschauwer *et al.* [12], while Venkataraman *et al.* [13] adapt to video QoE degradations by selecting one-hop, by-pass paths in overlay network that support application demands.

2.2 SDN-Based Approaches

SDN offers centralized control of data forwarding and has been used in recent approaches to optimized path assignment. SDN solutions are more light-weight and flexible than overlay networks, the former not depending on overlay structures.

Egilmez *et al.* [14] propose an analytical framework for dynamic routing of video traffic over QoS-optimized network paths. Unlike in the current Internet, where routes are not changed on a per-flow basis, SDN provides mechanisms for dynamic route management and calculation to meet different flow requirements (e.g., in terms of QoS). The authors mathematically formulate a constrained shortest path problem, for which the cost metric is based on packet loss and jitter. Focusing on scalable video coding, their approach supports QoS delivery of a video base layer, while enhancement layers can be assigned QoS-aware routes pending available capacity. OpenQoS, an SDN controller design based on dynamic QoS-driven routing that utilizes previously outlined optimization framework is described in [15]. Results have shown that OpenQoS outperforms existing approaches for RTP video streaming and HTTP adaptive streaming.

Jarschel *et al.* [16] present an SDN approach that utilizes different path selection schemes to enhance YouTube QoE. The most advanced scheme, application-aware path selection, leverages on application-level information about YouTube pre-buffered playtime to decide on a particular path. The actual path assignment is based on choosing one of the available links between two switches, whereas in contrast we will provide problem specification considering multi-hop paths.

In summary, a number of approaches have addressed path assignment with the goal of improving service quality. While most solutions tackle this problem from a QoS perspective, limited recent work (primarily focused on video streaming) has taken on a user-driven QoE perspective, relying on an understanding of the relationships between QoE and QoS. SDN is a viable approach in offering QoE-driven control of the path selection process, by providing an interface between application-level information and the network. Going beyond existing approaches, we propose a novel QoE-driven solution for the optimal routing of different service flows based on QoE models and user preferences. Previous work on path optimization has either neglected QoE aspects, or has assumed that all flows belonging to a session are routed along the same path between a given source and destination. We build on our generic approach proposed in [5] by formulating and solving the multi-user domain-wide QoE optimization problem.

3 The QoE-Driven Path Optimization Model

3.1 Model Overview

A high-level view of the previously proposed multimedia service delivery that leverages on the Q-POINT model and SDN [5] is given in Figure 1. It employs the Session Initiation Protocol (SIP) [17] to negotiate parameters for multimedia sessions that are to be established. The negotiation is assisted by an SIP application server with a QoS Matching and Optimization Function (QMOF) (introduced in our past work [18]), which calculates a set of configurations for each session that incorporate information such as feasible media flows, media codecs and bit rates, and user preferences in terms of favored media type(s). Session configurations include one optimal and several suboptimal configurations with regards to user- and service-imposed constraints (e.g., the configurations may differ in number of supported media flows and codec types). Calculated session and media flow parameters are passed to an SDN controller, which provides the obtained information to the Q-POINT optimization engine. The latter is run to determine which media flows should be routed along which paths in order to maximize the aggregated QoE. In its current design, Q-POINT focuses on planning flow routes by executing a single optimization process for multiple sessions, assumed to be entering the network, before they are established. We will extend our model so as to control the path optimization with regards to new sessions and flows being added and removed on a dynamic basis. The optimization output is finally translated into a set of forwarding rules, which are then installed

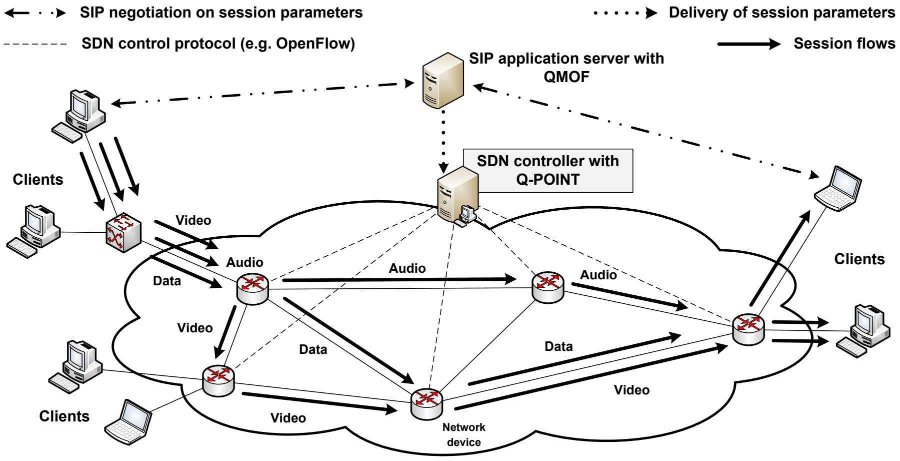
SIP negotiation on session parameters
Delivery of session parameters
SDN control protocol (e.g. OpenFlow)
Session flows
SIP application server with QMOF
SDN controller with Q-POINT
Clients
Clients
Clients
Video
Audio
Data
Audio
Audio
Video
Data
Data
Video
Video
Network device

Fig. 1. Multimedia service delivery based on Q-POINT and SDN

on network devices using, e.g., OpenFlow. Other technical and implementation-related specifics of the overall system, as well as its extensive evaluation and discussion on advantages it brings, will be presented as a part of future work.

When assigning a network path to a session flow, the optimization model implementation needs to specify all the constituent nodes and links for the given path. To achieve this objective, Q-POINT utilizes (a) an optimal configuration of each session that is to be established, (b) QoE-QoS mapping functions for different media types (e.g., for audio, video, and data), (c) network topology and link capacities, and (d) average end-to-end delay and packet loss probability.

We use a *session configuration* which includes information about media flows, such as their type, source and destination nodes, negotiated codec type and bit rate, minimum QoE value requested (which can be specified based on the chosen codec and bit rate for each flow), and weight factor, which indicates the importance of a flow within a session (e.g., audio being more important than video). In this work we will assume that audio flows belong to Voice over IP (VoIP)-based conversations, video flows to high-definition IPTV sessions, while data flows are generated by File Transfer Protocol (FTP)-based delivery.

As QoE is a multi-dimensional concept and is affected, among others, by session parameters and measurable QoS metrics, QoE models are used to capture the relationship between user-perceived quality and the considered influence factors. While other methods are possible, here we use the MOS metric with values on a scale from 1 to 5 in order to quantify quality for a media flow in the scope of the path selection process. For audio, the following model estimates MOS [19]:

$$MOS_{audio} = T - \alpha * p_{e2e} + \beta * d_{e2e} - \gamma * (d_{e2e})^2 + \delta * (d_{e2e})^3, \qquad (1)$$

where d_{e2e} and p_{e2e} are end-to-end (E2E) path delay and packet loss probability, respectively, while T, α, β, γ, and δ are function-specific values. T denotes maximum MOS value, specific for a chosen voice codec and bit rate, which is achievable when no packet loss and delay exist. All the chosen QoE models are representative parametric models for in-service MOS estimation. Q-POINT employs a parametric model that calculates video quality based on the video codec type (e.g., H.264), its bit rate, and E2E packet loss degradation [20]:

$$MOS_{video} = 1 + P(c_f, o_f) * \exp(-\frac{p_{e2e}}{Q(c_f)}), \tag{2}$$

where $P(c_f, o_f)$ and $Q(c_f)$ are model-specific functions of the codec type (c_f) and codec bit rate (o_f) to approximate influence of these parameters on MOS value. To assess QoE for the data transfer, the presented optimization model utilizes a logarithmic function that is described in [3]:

$$MOS_{data} = a * \log(b * o_f * (1 - p_{e2e})), \tag{3}$$

where o_f is average data traffic rate, while a and b are model-specific constants.

One of the key issues in the problem specification regards modeling network delay and packet loss probability. In this model, E2E delay for a given path considers propagation delay of the path's links and buffering delay of its "transit nodes", while average E2E loss probability takes into account loss at the path's transit nodes due to possible congestion (link loss is assumed zero). As values for link propagation delay are input parameters of the model, average buffering delay and loss probability in the nodes are calculated during the optimization process based on the incoming traffic rate, buffer configuration at a node, and link capacity. We assume that network nodes are configured to have one incoming buffer per each media type, i.e. one for audio, one for video, and one for data, while each buffer is modeled based on an $M/M/1/K$ queuing system. This allows us to calculate average delay and loss probability at node i as follows:

$$d_i = \frac{\frac{x}{b} * (1 + K * (\frac{x}{b})^{K+1} - (K+1) * (\frac{x}{b})^K)}{\frac{x}{e} * (1 - \frac{x}{b}) * (1 - (\frac{x}{b})^K)}, \tag{4}$$

$$p_i = \frac{(1 - \frac{x}{b}) * (\frac{x}{b})^K}{1 - (\frac{x}{b})^{K+1}}. \tag{5}$$

Parameter K represents the buffer size in number of packets, x overall incoming traffic rate for a specific buffer, e mean packet length, while b denotes the buffer processing rate (which corresponds to link capacity). While $M/M/1/K$ is a common way of modeling network node buffers, we note here that our aim is to extend Q-POINT so as to include other queuing system types and be able to approximate a wider range of traffic characteristics with regards to the inter-arrival time and service time distributions (e.g., assuming bursty traffic).

3.2 Mathematical formulation

We use the *generalized network flow model* with multi-commodity flows [21] to specify an integer linear optimization model as fast solution algorithms are available for such model type. The complete model notation is given in Table 1.

Table 1. Model notation

Model component	Symbol	Data type
Nodes	$N = \{i\}$, $\lvert N\rvert = n$	Integers
Links	$L = \{(i,j), i,j \in N\}$, $\lvert L\rvert = l$	Pairs of integers
Link delay, loss and capacity	$d_{ij} > 0$, $p_{ij} = 0$, $b_{ij} > 0$	Floats
Node delay and loss	$d_{iq} \geq 0$, $p_{iq} \geq 0$, $q \in \{1,2,3\}$	Floats
Multimedia sessions	$S = \{s\}$, $\lvert S\rvert = h$	Integers
Session MOS value	$u_s > 0$	Float
Media flows	$M = \{f\}$, $\lvert M\rvert = m$	Vectors of floats and integers
Flow source and destination	$src(f) = i^f_{src}$, $i^f_{src} \in N$, $dst(f) = j^f_{dst}$, $j^f_{dst} \in N$	Integers
Flow type, bit rate and codec	t_f, $o_f > 0$, c_f	String, float and string
Flow weight factor	$1 \geq w_f \geq 0$	Float
Flow MOS minimum	$v_f > 0$	Float
Flow MOS value	$u_f > 0$	Float
Node rates	$R_i = \{r^f_i = o_f : src(f) = i\}$, $i \in N$, $f \in M$	Vectors of floats

Let $G = (N, L)$ be a directed network specified by the set of *nodes* N and the set of *links* L (Figure 2). Each node $i \in N$ associates the cost per buffer $q \in \{1, 2, 3\}$ in terms of delay, d_{iq}, and loss probability, p_{iq}, which are calculated with functions (4) and (5), respectively. Each link $(i, j) \in L$ has the cost in terms of delay, d_{ij}, and it is assumed that the cost does not vary with the flow amount. Moreover, a link specifies capacity b_{ij}, the maximum flow amount on the link.

Let S be the set of h *multimedia sessions* that are to be established over network (N, L). Each session s may involve multiple *media flows.* A media flow

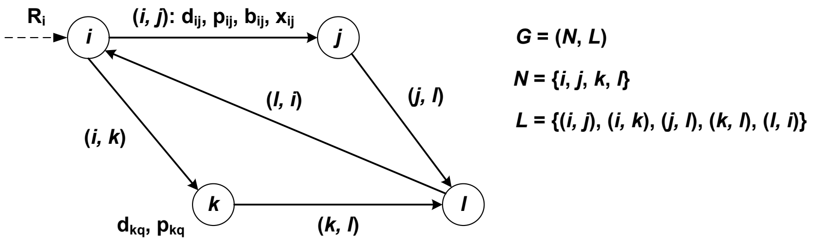
R$_i$
(i, j): d$_{ij}$, p$_{ij}$, b$_{ij}$, x$_{ij}$
i
j
(l, i)
(j, l)
(i, k)
k
l
d$_{kq}$, p$_{kq}$
(k, l)
G = (N, L)
N = {i, j, k, l}
L = {(i, j), (i, k), (j, l), (k, l), (l, i)}

Fig. 2. Network graph illustration for the path optimization problem

$f \in M$ is specified with its source $i^f_{src} \in N$, destination $j^f_{dst} \in N$, type t_f, codec c_f (if applicable, e.g., PCM for audio or H.264 for video), which also influences the accompanying mean packet size, bit rate o_f (e.g., 5 Mbit/s), weight factor in a session w_f, and defined MOS threshold, or minimum quality requested, v_f (e.g., 3.8 for audio with PCM and 80 kbps). MOS value for a flow, u_f, is predicted in the path selection process, based on formulas (1), (2) and (3) for a specific flow type, and then used to calculate MOS value for a session, u_s. Depending on session configurations, node $i \in N$ may be the source or the destination for multiple flows, or just act as a transit node on their paths. If i is the source for flow f, then node rate $r^f_i = o_f$, while R_i references rates of all the associated flows. If i is a transit node for flow f, then $r^f_i = 0$.

Model parameters While one group of the Q-POINT input parameters relates to session configurations, the second group encompasses MOS functions g, as given by equations (1), (2) and (3), which map application-level parameters (c_f, o_f) and network QoS parameters (d_{e2e}, p_{e2e}) to an MOS value. The last parameter group refers to network topology, which specifies how the nodes are interconnected and what are link characteristics (b_{ij}, d_{ij}).

Decision and auxiliary variables We choose two types of decision variables for this problem formulation: (a) x^f_{ij} denotes rate of flow $f \in M$ on link $(i, j) \in L$, which may be different from the original rate due to possible losses, and (b) $y^f_{ij} \in \{0, 1\}$ indicates whether link (i, j) is selected for the path of flow f or not.

If path loss probability p^f_{e2e} is calculated based on loss probability of each node (p_{iq}) on the path, then the derived loss formula incorporates a product of decision variables y^f_{ij} (to select network segments that contribute to the E2E loss), which makes the mathematical formulation non-linear. Figure 3 illustrates the applied solution to this issue by introducing a virtual network node Z and an auxiliary variable z^f_i. Node Z represents the sink for packets being lost at the path's nodes, while z^f_i holds loss rate of flow f at node i (if packet loss occurs). All network nodes $i \in N$ are connected to Z with virtual links, which are characterized by $b_{iZ} = \infty$, delay $d_{iZ} = 0$, and loss probability $p_{iZ} = 0$.

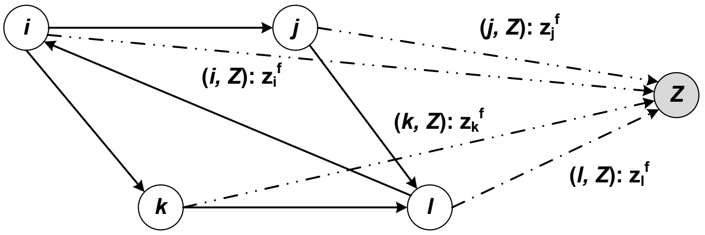

Fig. 3. Network graph with virtual node Z as the "lost packets' sink"

Building on the applied solution, average loss probability for flow $f \in M$ can then be calculated as the sum of loss rates at nodes that are included in the flow's path over original flow rate (note that the path loss is not additive with respect to node loss probability, p_{iq}):

$$p^f_{e2e} = \frac{\sum_{i \in N: i \neq dst(f)} z^f_i}{o_f}; \qquad z^f_i = \sum_{\substack{\{i \in N: \\ i = tail(i,j), (i,j) \in L\}}} x^f_{ij} * \frac{p_{iq}}{1 - p_{iq}}. \tag{6}$$

To calculate E2E delay for flow $f \in M$, on the other hand, delays on each node and link of the flow's path are summed up:

$$d^f_{e2e} = \sum_{\substack{\{(i,j) \in L: \\ i = tail(i,j), i \neq dst(f), \\ i \in N\}}} y^f_{ij} * (d_{iq} + d_{ij}). \tag{7}$$

Objective function As per the problem specification, the Q-POINT objective is to maximize the sum of MOS values over all multimedia sessions:

$$maximize \sum_{s \in S} u_s, \tag{8}$$

where MOS value for a session is calculated as a weighted sum of MOS values for the comprising (one or more) media flows:

$$u_s = \sum_{\{f \in M: session(f) = s\}} w_f * u_f. \tag{9}$$

Model constraints Table 2 depicts mathematical formulation of the model constraints. The *Minimum MOS* constraint forces Q-POINT to select a path that provides, at least, MOS value v_f for flow f, thus satisfying minimum quality requirements for a specific flow and also guaranteeing a certain fairness among all end-users. *Maximum link rate* denotes that link $(i, j) \in L$ can admit flow f only if its link rate x^f_{ij} does not exceed the link capacity. Similarly, the *Maximum sum of link rates* constraint imposes that the sum of link rates for flows following the same link cannot exceed the link capacity. *Maximum link rate* is specified so as to simplify the mathematical formulation and facilitate the problem solving.

The *Flow conservation* constraint for each flow specifies that incoming link rate x_{ij} at node j is divided between outgoing link rate x_{jk} and loss rate z^f_j. If j is the source for flow f, then incoming link rate equals to node rate r^f_j. *Link selection* forces a flow to follow only one outgoing link from its source, one incoming and one outgoing link at a transit node, and only one incoming link at the flow's destination. This means that flows are non-splittable and cannot use concurrent paths to reach their destinations, leading to a complex-to-solve model. Finally, the *Loop-back links* constraint requires Q-POINT to avoid choosing links that would send flows back towards their source nodes.

Table 2. Model constraints

Model constraint	Mathematical formulation
Minimum MOS	$u_f \geq v_f \ , \forall f \in M$
Maximum link rate	$x_{ij}^f \leq y_{ij}^f * b_{ij} \ , \forall f \in M \ , \forall (i,j) \in L$
Maximum sum of link rates	$\sum_{f \in M} x_{ij}^f < b_{ij} \ , \forall (i,j) \in L$
Flow conservation	$\sum_{\{k:k=head(j,k)\}} x_{jk}^f + z_j^f = r_j^f \ ,$ $\forall f \in M \ , \forall j \in N : j = src(f)$
	$\sum_{\{k:k=head(j,k)\}} x_{jk}^f + z_j^f - \sum_{\{i:i=tail(i,j)\}} x_{ij}^f = 0 \ ,$ $\forall f \in M \ , \forall j \in N : (j \neq src(f)) \wedge (j \neq dst(f))$
Link selection	$\sum_{\{k:k=head(j,k)\}} y_{jk}^f = 1 \ ,$ $\forall f \in M \ , \forall j \in N : j = src(f)$
	$\sum_{\{k:k=head(j,k)\}} y_{jk}^f - \sum_{\{i:i=tail(i,j)\}} y_{ij}^f = 0 \ ,$ $\forall f \in M \ , \forall j \in N : (j \neq src(f)) \wedge (j \neq dst(f))$
	$\sum_{\{i:i=tail(i,j)\}} y_{ij}^f = 1 \ ,$ $\forall f \in M \ , \forall j \in N : j = dst(f)$
Loop-back links	$y_{ij}^f + y_{kl}^f \leq 1 \ , \forall f \in M \ ,$ $\forall (i,j) \in L, \forall (k,l) \in L : (((i,j) > (k,l)) \wedge (i = l) \wedge (j = k))$

4 Model Implementation

We use the IBM Optimization Programming Language [22] to formulate our model. One of the major issues regarding model implementation was the existence of non-linear functions in the initial formulation (e.g., equations (4) and (5) incorporate decision variable x_{ij}). Equation (7) for calculating E2E delay on a path includes a non-linear product of binary variable y_{ij} and continuous function d_{iq}. To linearize it, we apply a technique that introduces substitute decision variables. In this case, d_{iq} is defined as a new decision variable, which in turn creates a product of binary and continuous variable. The latter product can be replaced by a new continuous decision variable, which we refer to as yd_{ijq}. To be able to employ this substitution, additional constraints need to be defined:

$$yd_{ijq} \geq 0, \quad yd_{ijq} \leq y_{ij} * M, \quad yd_{ijq} \leq d_{iq}, \quad yd_{ijq} \geq d_{iq} - (1 - y_{ij}) * M, \quad (10)$$

where M is the upper bound on value of d_{iq}. Similarly, equation (6) includes a product of continuous variable x_{ij} and continuous function p_{iq}. This product is linearized by using the one-dimensional method from [23], which introduced five continuous decision variables, two binary decision variables and several of the accompanying constraints to our initial formulation.

5 Model Evaluation

In this section, we present an initial evaluation of the proposed model. The evaluation examines the problem solving time with respect to different number of flows to be routed and network topologies. It also analyzes overall QoE gains of Q-POINT over the shortest path approach typically used in current networks.

We use IBM ILOG CPLEX Optimization Studio 12.5 [22], CPLEX Optimizer's mixed integer solver and the branch-and-cut algorithm with default settings. The solver is run in Debian Linux 6.0.8 on a workstation with an Intel Xeon CPU @ 2.6 GHz and 32 GB of RAM. To obtain numerical results for the evaluation, m flow requests are generated: $\frac{2m}{5}$ audio flows, $\frac{2m}{5}$ video flows, and $\frac{m}{5}$ data flows. Flows of the same type are generated with the same characteristics (Table 3). For each flow we randomly choose its source and destination, but in a way that each network node serves as the source and the destination to a similar portion of m flows. After flow generation, the Q-POINT model is run.

The evaluation network topologies are shown in Figure 4. The first one is a random topology, while the other one is modeled against the Croatian National and Research Network (CARNet), i.e. a part of its core network. For both topologies capacity of each link is set to 1 Gbps, while link delay is randomly chosen from {10 ms, 20 ms, 30 ms}. Each network node is pre-configured with 3 buffers. Audio buffer size is set to 1000 bytes (i.e., 5 audio packets), video buffer size to 28800 bytes (i.e., 20 packets), while data buffer size is set to 30000 bytes. The weighted fair queuing discipline is assumed at network nodes, which all serve as source and destination to flows of different type, with buffer weights set to 0.3, 0.5 and 0.2 for audio, video and data flows, respectively. The chosen network values were empirically derived and impact of their variations on the optimization result will be analyzed in future work, as well as impact of more complex network topologies. All results are obtained over 10 test runs for each topology and flow number m, which is specified from {100, 200, 300, 400, 500}.

With respect to the CARNet-like topology, Table 4 shows the sum of QoE values over all flows for Q-POINT and the shortest path selection, which is based on the "hop-count" metric. While Q-POINT achieves higher aggregate QoE for each m value, a notable difference occurs for $m = 500$, when overall traffic increases link utilization considerably (for some links to above 50%). Our model consequentially aims to distribute video and data paths so as to "balance" traffic load per node, thus minimizing QoE degradations. For $m = 500$, Q-POINT

Table 3. Flow characteristics for the Q-POINT evalution

Flow type	**Codec**	**Bit rate** [Mbps]	**Maximum MOS**	**Mean packet length** [bytes]	**Generated no. of flows**
Audio	PCM	0.08	4.3	200	$\frac{2m}{5}$
Video	H.264	5.0	4.7	1440	$\frac{2m}{5}$
Data	-	5.0	4.5	1500	$\frac{m}{5}$

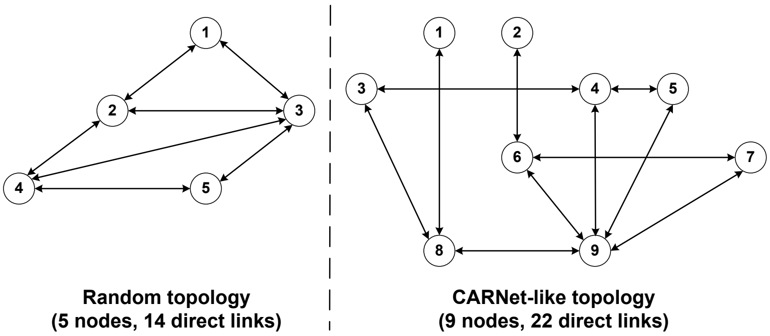

Fig. 4. Network topologies for the Q-POINT evaluation

obtains, e.g., video loss probability under 0.15% at each node, with all video flows over two-hop paths facing loss probability of 0.23% on average and achieving MOS of 4.56 on average. With the same flow configuration, the shortest path selection results in, e.g., video loss probability at node 9 of 0.87%. Moreover, 66 video flows are assigned two-hop routes with loss probability of 1.12% on average, leading to their average MOS value of 3.99. Although this preliminary evaluation shows some encouraging results on QoE gains of Q-POINT over the shortest path, a thorough analysis needs to be performed to derive general conclusions.

Table 4. Comparison of the sum of QoE values over all flows

CARNet-like topology	$m = 100$	$m = 200$	$m = 300$	$m = 400$	$m = 500$
Shortest path selection	448.5	892.9	1330.3	1754.2	**2147.4**
Q-POINT model	449.0	895.6	1338.8	1775.1	**2200.7**

Average execution time of the solver is shown in Figure 5. While Q-POINT yields acceptable performance for the random network, which is of a simpler topology than the CARNet-like network, and $m = \{100, 200\}$, it is evident that running a single optimization of paths for that many flows in a network would be too time consuming to apply the model for dynamic network reconfiguration.

6 Conclusion

In this paper, we have presented Q-POINT, a QoE-driven path optimization model for multimedia services. In contrast to traditional networks, where flows with the same destination typically follow the same path, Q-POINT calculates the best path for each service flow so as to maximize the aggregated QoE for a whole network domain. The key contribution of this paper is the presented mathematical model, which is formulated as a mixed integer linear program.

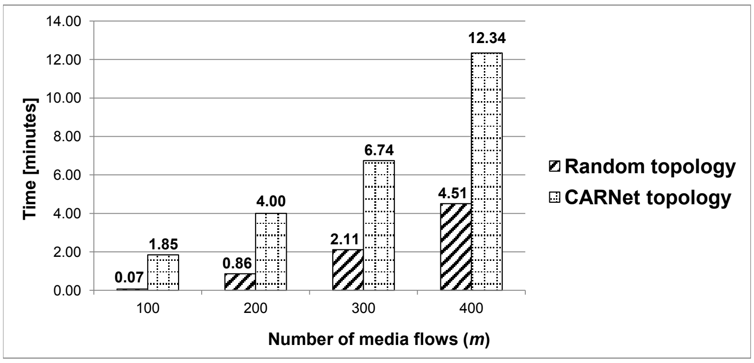

Fig. 5. The solver execution time

The preliminary evaluation shows that our model increases the overall QoE, which means that end-users will be more satisfied with the delivered service.

Our work opens up several interesting research aspects. First is to evaluate impact of different QoE-QoS mapping functions on resource utilization and of using multi-path transfer, with the latter simplifying the model complexity since flows become splittable. As end-users frequently establish new sessions and flows, we are currently extending Q-POINT to control optimization for given traffic dynamics, while trying to keep the number of path reconfigurations for the existing flows to a minimum. We also plan to address the applicability and benefits of our approach in the context of additional service types (e.g., adaptive video streaming over HTTP and on-line gaming) and more complex traffic mixes. A step further will be to explore heuristics that will allow us to achieve a satisfactory QoE result in minimal (or acceptable) execution time. Finally, we have also started to implement the model within the SDN framework by developing an SDN controller application to run Q-POINT.

Acknowledgments. This work has been supported by the European Community's 7th Framework Programme under grant agreement no. 285939 (ACROSS), and the research projects 036-0362027-1639 and 071-362027-2329 funded by the Ministry of Science, Education, and Sports of the Republic of Croatia. Additional support was provided by the KK-stiftelsen under the READY project.

References

1. Schatz, R., Hoßfeld, T., Janowski, L., Egger, S.: From Packets to People: Quality of Experience as New Measurement Challenge. In: Biersack, E., Callegari, C., Matijasevic, M. (eds.) Data Traffic Monitoring and Analysis. LNCS, vol. 7754, pp. 219–263. Springer, Heidelberg (2013)
2. Thakolsri, S., Khan, S., Steinbach, E. G., Kellerer, W.: QoE-Driven Cross-Layer Optimization for High Speed Downlink Packet Access. J. Communications, North America. 4, 669–680 (2009)

3. Brajdic, A., Kassler, A., Matijasevic, M.: Quality of Experience based Optimization of Heterogeneous Multimedia Sessions in IMS. In: 2011 BCFIC, pp. 25–32. (2011)
4. Open Networking Foundation: Software-Defined Networking. White paper, (2012)
5. Kassler, A., Skorin-Kapov, L., Dobrijevic, O., Matijasevic, M., Dely, P.: Towards QoE-driven Multimedia Service Negotiation and Path Optimization with Software defined Networking. In: 20th SoftCOM, pp. 1–5. (2012)
6. Wang, Z., Crowcroft, J.: Quality-of-service Routing for Supporting Multimedia Applications. IEEE J. Selected Areas in Communications. 14, 1228–1234 (1996)
7. Lorenz, D. H., Orda, A.: QoS Routing in Networks with Uncertain Parameters. IEEE/ACM Trans. on Networking. 6, 768–778 (1998)
8. Kumar, D., Kashyap, D., Mishra, K. K., Mishra, A. K.: Routing Path Determination Using QoS Metrics and Priority based Evolutionary Optimization. In: 13th IEEE HPCC, pp. 615–621. (2011)
9. Kuipers, F., Van Mieghem, P., Korkmaz, T., Krunz, M.: An Overview of Constraint-based Path Selection Algorithms for QoS Routing. IEEE Communications Magazine. 40, 50–55 (2002)
10. Lu, T., Zhu, J.: A Genetic Algorithm for Finding a Path Subject to Two Constraints. Applied Soft Computing. 13, 891–898 (2013)
11. Amram, N., Fu, B., Kunzmann, G., Melia, T., Munaretto, D., Randriamasy, S., Sayadi, B., Widmer, J., Zorzi, M.: QoE-based Transport Optimization for Video Delivery over Next Generation Cellular Networks. In: 2011 IEEE ISCC, pp. 19–24. (2011)
12. De Vleeschauwer, B., De Turck, F., Dhoedt, B., Demeester, P., Wijnants, M., Lamotte, W.: End-to-end QoE Optimization Through Overlay Network Deployment. In: 2008 ICOIN, pp. 1–5. (2008)
13. Venkataraman, M., Chatterjee, M.: Effects of Internet Path Selection on Video-QoE: Analysis and Improvements. IEEE/ACM Trans. on Networking, 14 pp. (2013)
14. Egilmez, H. E., Civanlar, S., Tekalp, A. M.: An Optimization Framework for QoS-enabled Adaptive Video Streaming over OpenFlow Networks. IEEE Trans. on Multimedia. 15, pp. 710–715 (2013)
15. Egilmez, H. E., Dane, S. T., Bagci, K. T., Tekalp, A. M.: OpenQoS: An OpenFlow Controller Design for Multimedia Delivery with End-to-end Quality of Service over Software-Defined Networks. In: 2012 APSIPA, pp. 1–8. (2012)
16. Jarschel, M., Wamser, F., Hoehn, T., Zinner, T., Tran-Gia, P.: SDN-based Application-Aware Networking on the Example of YouTube Video Streaming. In: 2nd EWSDN, pp. 87–92. (2013)
17. Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., Schooler, E.: SIP: Session Initiation Protocol. IETF (2002)
18. Skorin-Kapov, L., Matijasevic, M.: Modeling of a QoS Matching and Optimization Function for Multimedia Services in the NGN. In: 12th IFIP/IEEE MMNS, pp. 55–68. (2009)
19. Moura, N., Vianna, B., Albuquerque, C., Rebello, V., Boeres, C.: MOS-Based Rate Adaption for VoIP Sources. In: 2007 IEEE ICC, pp. 628–633. (2007)
20. Yamagishi, K., Hayashi, T.: Parametric Packet-Layer Model for Monitoring Video Quality of IPTV Services. In: 2008 IEEE ICC, pp. 110–114. (2008)
21. Ahuja, R. K., Magnanti, T. L., Orlin, J. B.: Network Flows: Theory, Algorithms, and Applications. Prentice Hall, New Jersey (1993)
22. IBM ILOG CPLEX Optimization Studio, `http://www-03.ibm.com/software/products/us/en/ibmilogcpleoptistud/`
23. D'Ambrosio, C., Lodi, A., Martello, S.: Piecewise Linear Approximation of Functions of Two Variables in MILP Models. Operat. Research Letters. 38, 39–46 (2010)