Ant Colony Optimization for QoE-Centric Flow Routing in Software-Defined Networks

Ognjen Dobrijevic, Matija Santl, and Maja Matijasevic
University of Zagreb, Faculty of Electrical Engineering and Computing
Unska 3, 10000 Zagreb, Croatia
Email: {ognjen.dobrijevic, maja.matijasevic}@fer.hr

***Abstract*—We present design, implementation, and an evaluation of an ant colony optimization (ACO) approach to flow routing in software-defined networking (SDN) environments. While exploiting a global network view and configuration flexibility provided by SDN, the approach also utilizes quality of experience (QoE) estimation models and seeks to maximize the user QoE for multimedia services. As network metrics (e.g., packet loss) influence QoE for such services differently, based on the service type and its integral media flows, the goal of our ACO-based heuristic algorithm is to calculate QoE-aware paths that conform to traffic demands and network limitations. A Java implementation of the algorithm is integrated into SDN controller OpenDaylight so as to program the path selections. The evaluation results indicate promising QoE improvements of our approach over shortest path routing, as well as low running time.**

I. Introduction and Motivation

Quality of experience (QoE) [1] represents a shift in analyzing service performance from the end-user perspective, which goes beyond predominantly *network-related parameters* (referred to as quality of service, QoS) to include diverse technical and non-technical factors that impact the user QoE. Aside from network parameters, QoE involves (a) *application-related parameters*, e.g. content type and available formats, (b) *system-related parameters*, such as features of end-user equipment, (c) *user-related parameters*, e.g. end-user preferences, and (d) *context-related parameters*, such as service cost.

The QoE expectations largely depend on the type of service that an end-user is accessing, its application flows and their network resource demands. For instance, while a conversational audio generally calls for one-way delay of less than 150 ms and packet loss rate under 1%, a high-definition video requires a route with throughput of, e.g., 2-5 Mbit/s and the loss rate up to several percent [2]. By monitoring relevant impact factors (IFs) and feeding their values into *QoE models*, which are functions for estimating a QoE measure, configuration of end-user services and the underlying network can be customized so as to achieve good end-user QoE. The latter falls under the umbrella of *QoE-centric service and network management* [3].

With the software-defined networking (SDN) paradigm [4], which offers (a) decoupling of the network control and data planes, (b) a global network state view by SDN controllers, and (c) common interfaces between applications and SDN controllers, we are able to develop a routing model that:

(1) exploits (tight) cooperation between end-user applications and the network control on collecting QoE IF values,
(2) arbitrates among competing application flows in the event of network congestion by considering their type, and
(3) is able to meet demands on the data forwarding configuration more efficiently than shortest path algorithms.

Motivated by this potential, we seek to advance from shortest path routing (SPR) to finding paths in a manner that exploits the respective QoE models, i.e. *QoE-centric routing*. The overall goal is to calculate the "best available" paths for different multimedia service types and their flows, subject to traffic demands and network constraints, in order to maximize end-user QoE. Since QoE generally depends on multiple QoS metrics in a non-linear fashion (e.g., cf. [5]), while a multi-constrained routing problem is known to be NP-complete [6], we develop a meta-heuristic algorithm based on ant colony optimization (ACO) that calculates QoE-aware paths. In this respect, the main contribution of this paper consists of:

(1) design of an ACO-based heuristic algorithm for flow routing in software-defined networks that employs different QoE models (thus achieving a QoE-centric routing),
(2) implementation of the proposed algorithm in Java and its integration into SDN controller OpenDaylight [7], and
(3) an algorithm evaluation against SPR (as the latter is commonly used in current communication networks and dominantly implemented in SDN controllers), with promising QoE improvements over SPR and low running time.

The rest of the paper is organized as follows. Section 2 outlines the related work with respect to QoS-/QoE-driven routing and SDN, while Section 3 presents design of our QoE-centric path selection model and algorithm. An evaluation of the ACO-based algorithm is described in Section 4, followed by the conclusion and future work overview.

II. Related Work

In recent years there are notable SDN research results which deal with models and platforms for efficient development of routing services (e.g. [8], [9]), but there is still the need for a routing model that examines various IFs and maximizes user QoE by employing a QoE measure as the routing metric.

A framework for QoS-optimized video routing is presented in [10]. The routing problem is solved as a constrained shortest path model, which structures the metric as a weighted sum of packet loss and jitter. An extension to the previous work, which designs a distributed SDN control plane for QoS provisioning over multiple domains, is described in [11]. Therein, the QoS routing problem is solved by the Lagrangian relaxation based aggregated cost algorithm. Agarwal *et al.* in [12] propose to solve routing optimization problem by utilizing the fully

978-3-901882-77-7 © 2015 IFIP

polynomial time approximation scheme and finding paths that minimize the maximum utilization of network links.

QoE Fairness Framework for adaptive video streaming (AVS) is introduced in [13]. It fairly maximizes the amount of network resources for contending clients by using QoE models that "map" video bit rates to QoE values, running a bandwidth allocation algorithm, and calculating an optimal set of video rates for all clients. A solution exploiting SDN and multi-protocol label switching to improve QoE for AVS is described in [2]. This approach measures application-related QoE IFs, such as video start-up latency and buffering rate, and executes a constrained shortest path first algorithm in the cases of QoE degradation so as to select a better route. Jarschel *et al.* propose path selection in access networks that is based on monitoring video buffer level and stalling manifestations for YouTube [14].

To the best of our knowledge, the routing model described in this paper differs from the recent results in that it considers different QoE models, relating to diverse service and flow types, and employs QoE measures for routing.

III. QoE-Centric Routing Based on Ant Colony

Figure 1 illustrates the overall concept of our QoE-centric flow routing approach. The concept is motivated by the economic traffic management [15], which calls for cooperation between network and service providers – the main incentive for network providers is an efficient resource utilization, while service providers seek to maximize QoE for their end-users.

A. QoE-Centric Flow Routing Model

We build our QoE-centric routing model on the cooperation among several components to measure and collect QoE IF values in an SDN network, namely end-user clients and media content servers (symbolized with a video delivery server in Fig. 1), SDN applications, and SDN controllers. In particular, the clients and the content servers supply SDN applications with values of relevant QoE IFs (i.e. *QoE reporting*), while SDN applications prepare a needed input for SDN controllers and the path selection process. The complete model design, which can be applied to any multimedia service, is out of scope of this paper. Therefore, we highlight the involved components

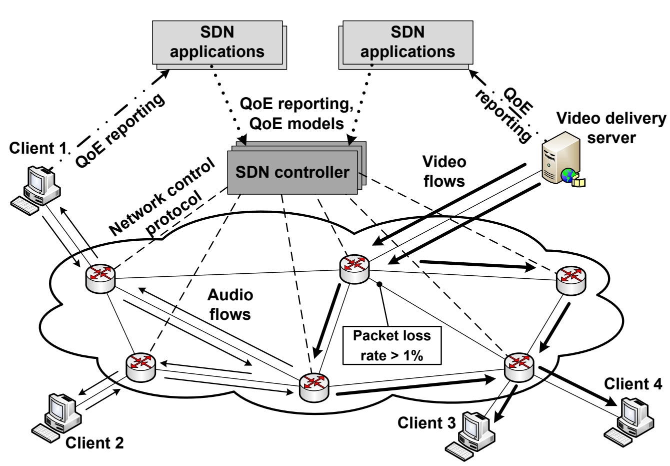

Fig. 1. QoE-centric flow routing model for software-defined networks

(illustrating only one SDN controller) and their purpose in a scenario of configuring network paths while establishing video service sessions between end-user clients and a server [16]:

(1) clients (e.g., *Clients 3* and *4* in Fig. 1) issue content requests to the *Video delivery server*, which include end-user preferences (e.g. on video resolution) and equipment features (e.g., screen resolution and supported codecs);
(2) the *Video delivery server* conveys service parameters (e.g., available video codecs, resolutions and bit rates) and server features, along with the client information, to an SDN application with the "parameter matching function";
(3) after collecting all the IF values, the SDN application matches them to determine feasible delivery parameters for each client [17], such as video codecs and bit rates;
(4) the SDN application delivers video session parameters to an associated SDN controller, along with the service QoE model supplied by the respective service provider;
(5) having a complete view on the underlying network (including packet loss rate and delay estimation per device), which is retrieved by, e.g., OpenFlow [18] and from monitoring tools, the controller runs the QoE-centric routing algorithm and programs the needed network paths.

B. ACO-Based Heuristic Algorithm

We solve the routing problem on a weighted graph with a QoE measure as the ultimate metric. QoE inevitably depends on QoS, which is generally expressed in terms of delay, jitter and packet loss rate, and often combines multiple QoS metrics which all need to be satisfied in order to achieve good QoE. Since the QoE-QoS relationship is of a non-linear nature (e.g., cf. [5]) and multi-constrained routing is an NP-complete problem [6], we design an ACO-based meta-heuristic algorithm so as to find the QoE-centric routing solutions. While being a convenient technique for selecting paths that can maximize QoE, subject to traffic demands and network constraints, ACO was also chosen as it can offer a favorable trade-off between diversification and intensification of the solution space.

ACO [19] uses software *ants*, which traverse network graph towards a given destination and, after finding a path, leave *pheromone trails* on the way back to their source. The *pheromones* influence the ACO's stochastic nature of building solutions, where node with a stronger trail is more likely to be a final solution element. However, over time the pheromones evaporate and only nodes that are visited by different ants more frequently have their pheromone density increased. In our model, for a new incoming media flow SDN controller takes a snapshot of the underlying network performance and then runs the algorithm, which simulates ants searching for paths through the graph corresponding to the network topology.

We associate weights to the graph nodes based on values of delay and packet loss rate for each network device ($delay[node]$ and $lossRate[node]$, respectively). QoE for a flow is estimated based on the flow type and its QoE model, for which delay and packet loss rate of a path (i.e. end-to-end, E2E) need to be calculated. Since delay is an additive metric, E2E delay (d_{e2e}) sums up delay of each node on a path [20]:

$$d_{e2e} = \sum_{node \in path} delay[node]. \quad (1)$$

However, packet loss rate is a multiplicative metric and E2E loss (l_{e2e}) regards loss rates on a path as follows [20]:

$$l_{e2e} = 1 - \prod_{node \in path} (1 - lossRate[node]). \quad (2)$$

As our design (presented by Algorithm 1) assumes different multimedia services and flow types (non-exhaustive list includes audio, video and data), for each flow type we specify one ant type. More ant types can easily be added to involve new flow types. Multiple ants of the same type are then sent from the flow source to its destination in several iterations, tracking estimated QoE and seeking to maximize the final result.

When ants explore the graph, each of them keeps a list of the nodes it has already visited (also used when QoE for a path is computed). To determine which neighboring node to visit next, an ant calculates the probability for each adjacent node in the following way (visit node j from node i):

$$p_{ij} = \frac{\tau_{ij}^{\alpha} \cdot \eta_{ij}^{\beta}}{\Sigma_k \tau_{ik}^{\alpha} \cdot \eta_{ik}^{\beta}}. \quad (3)$$

In eq. (3), τ_{ij} represents the pheromone trail between nodes i and j, η_{ij} represents the nearness value between nodes i and j (which is influenced by their delay and loss rate), while parameters α and β control how much the pheromones and the nearness values affect the probability of node j being chosen.

Algorithm 1 ACO-Based Algorithm for QoE-Centric Routing

1: Set algorithm parameters (α, β, $antPopulation$, $evaporationRate$, $numberOfIterations$)
2: Initialize pheromone trails ($numberOfNodes$, $nodesDelay$, $nodesLossRate$)
3: **procedure** SENDANT(ant)
4: $currNode$:= $srcNode$
5: ant.AddToPath($currNode$)
6: **while** $currNode \neq dstNode$ **do**
7: $probabilities$:= CalculateProbability($currNode$)
8: $nextNode$:= SelectNode($probabilities$)
9: ant.AddToPath($nextNode$)
10: $currNode$:= $nextNode$
11: **end while**
12: **end procedure**
13: **procedure** RUN($flowType$)
14: **for** k in $1..numberOfIterations$ **do**
15: **for** i in $1..antPopulation$ **do**
16: ant := CreateAnt($flowType$)
17: SendAnt(ant)
18: ant.CalculateQoEValue()
19: $ants$.Append(ant)
20: **end for**
21: EvaporatePheromones()
22: $totalQoEValue$:= Σ_{ant} ant.GetQoEValue()
23: **for** ant in $ants$ **do**
24: EnhancePheromones(ant.GetPath(), ant.GetQoEValue() / $totalQoEValue$)
25: **end for**
26: **end for**
27: $bestAnt$:= $argmax_{ant}$ (ant.GetQoEValue())
28: **return** $bestAnt$.GetPath()
29: **end procedure**

The algorithm comprises several phases. After specification of its parameters α, β, $antPopulation$, $evaporationRate$, and $numberOfIterations$ (line 1 in Algorithm 1), which can be tuned to a particular SDN environment, the next phase is initialization of the pheromones and the nearness values (line 2). The latter values are populated based on $lossRate[node]$ and $delay[node]$. The main algorithm procedure is $Run()$ (lines 13 to 29), which, given the flow type, creates $antPopulation$ ants in each of the $numberOfIterations$ iterations and computes a path for every ant.

After $antPopulation$ ants have crossed the graph (lines 15 to 20), the pheromone trails are updated. First, all trails are evaporated with rate $evaporationRate$ (line 21). Then, for each ant that has found a path from $srcNode$ to $dstNode$, the pheromone trails on the path are enhanced relative to the path QoE value (line 24). After the algorithm runs for $numberOfIterations$, the ant path with the greatest QoE value is returned as the solution for the given flow (line 28).

IV. ALGORITHM EVALUATION IN AN SDN PROTOTYPE

This section presents an initial evaluation for the presented algorithm, which is conducted in an SDN prototype. The prototype includes Mininet [21], which emulates an SDN network, and SDN controller OpenDaylight (holding a Java implementation of our algorithm), both software running on commodity desktop computers connected to a Gigabit Ethernet network. The evaluation goal is to compare the ACO-based algorithm against SPR, in particular the latter's OpenDaylight realization, with regards to achieved QoE and the time needed to calculate paths (i.e. running time).

The evaluation scenario includes SDN controller selecting paths for emulated media flows in two networks (Figure 2), each with the specific network congestion conditions that are chosen as illustrative examples (simpler network topology is based on [2]). We emulate three flow types with typical parameters, namely a conversational audio (rate 80 kbps, encoding PCM), an IP television video (rate 5 Mbps, encoding H.264) and a file transfer data (average rate 1 Mbps), all flows having $srcNode = 1$ and $dstNode = 3$.

Without loss of generality, for the evaluation we use particular QoE estimation models. The models express QoE on a Mean Opinion Score scale from '1' (the "worst quality") to '5' (the "best quality"). In the audio QoE model [22]:

$$QoE_{audio} = T - \gamma \cdot l_{e2e} + \delta \cdot d_{e2e} - \epsilon \cdot (d_{e2e})^2 + \zeta \cdot (d_{e2e})^3, \quad (4)$$

T denotes maximum QoE value achievable when no loss and delay exist (4.3 for standard PCM), while γ, δ, ϵ, and ζ are model-specific values. Video QoE is calculated as [23]:

$$QoE_{video} = 1 + R(c_v, o_v) \cdot \exp(-\frac{l_{e2e}}{Q(c_v)}), \quad (5)$$

where $R()$ and $Q()$ are model-specific functions of video codec type (c_v) and bit rate (o_v), while maximum QoE for the chosen video flow parameters and no loss is 4.53. For data QoE [24]:

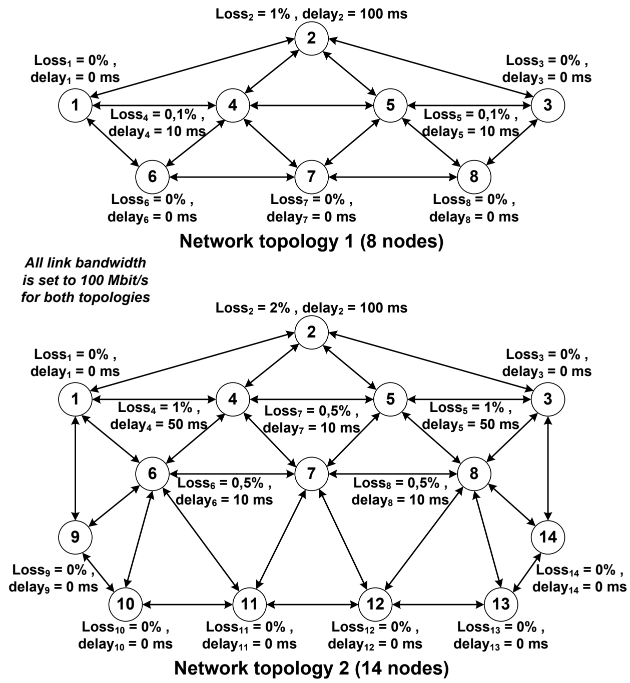

Fig. 2. SDN networks emulated in Mininet

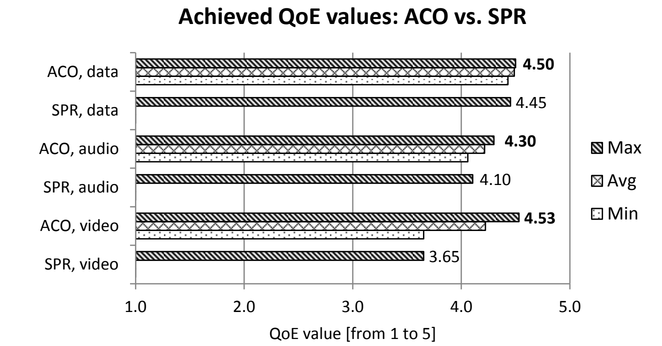

Fig. 3. Comparison of the achieved QoE values (for *Network topology 1*)

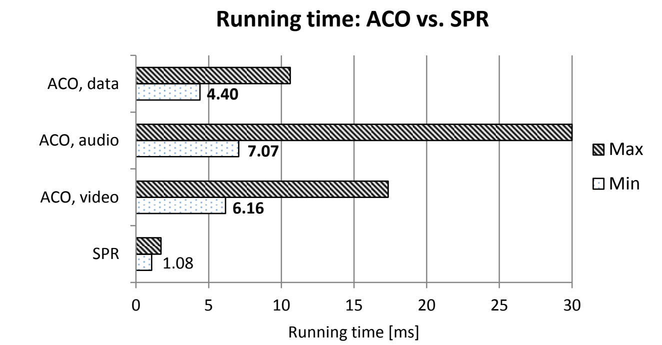

Fig. 4. Comparison of the running times (for *Network topology 2*)

$$QoE_{data} = \lambda \cdot \log(\mu \cdot o_d \cdot (1 - l_{e2e})), \quad (6)$$

o_d is average bit rate, λ and μ are model-specific constants, while maximum QoE under no loss is set to 4.5 (based on [24]). In the tests, we change the ACO parameters among the following experimental values: $numberOfIterations \in \{5, 10, 15, 20, 25\}$, $antPopulation \in \{5, 10, 15, 20, 25\}$ and $evaporationRate \in \{0.4, 0.5, 0.6, 0.7\}$, and different value combinations *(*$numberOfIterations$, $antPopulation$, $evaporationRate$*)* are applied. For each of the emulated SDN networks and parameter value combinations, 10 tests are run.

Figure 3 shows a comparison of the measured QoE values for *Network topology 1* (due to the space limit, results for *Network topology 2* are omitted). Since QoE values are obtained in respect to different combinations of the ACO parameter values, we summarize the results for our algorithm with the minimal, average, and maximal QoE values over all the combinations applied. The comparison demonstrates that the ACO-based algorithm outperforms SPR and, for the given network topologies and conditions, is able to find paths with the maximum QoE values. Specifically with regards to video, for *Network topology 1* the maximal QoE value obtained by ACO is 24.1% higher than the SPR's value, while for *Network topology 2* the ACO's maximal QoE value of 4.06 ("good quality") exceeds the SPR's value of 2.99 ("poor quality"). Although these results are promising for our algorithm, further analysis is needed to achieve more general conclusions.

Obtained for *Network topology 2*, Figure 4 shows the minimal and maximal running times of the ACO-based and SPR algorithms. This evaluation indicates that our algorithm runs on a time scale similar to SPR and does not induce a significant additional overhead. The running times are even more interesting when we consider overall programming network time, i.e. the time needed both to compute a path and configure it on every network device along the way. For instance, results for *Network topology 2* show that, regardless of using the ACO-based algorithm or SPR, the minimal programming network time is above 200 ms. This makes running time of our algorithm almost a negligible part of the overall time consumption, which requires other approaches to be dealt with.

V. Conclusions and Future Work

This paper presents design and an evaluation of a quality of experience (QoE)-centric approach to flow routing in software-defined networks, which is based on ant colony optimization. The evaluation results indicate promising QoE improvements over shortest path routing, as well as low running time.

For the scalability purposes, we are currently developing a prototype with different SDN controllers being responsible of configuring paths for respective categories of multimedia services. Our future work will involve human subjects evaluating the proposed approach against other routing solutions and for different multimedia services (e.g., adaptive video streaming), as well as for different network sizes and conditions.

Acknowledgment

The research leading to these results has received funding from the project "Information and communication technology for generic and energy-efficient communication solutions with application in e-/m-health (ICTGEN)" co-financed by the European Union from the European Regional Development Fund.

REFERENCES

[1] European Network on Quality of Experience in Multimedia Systems and Services (COST Action IC 1003), "Qualinet White Paper on Definitions of Quality of Experience (2012)," Patrick Le Callet, Sebastian Möller and Andrew Perkis, eds., Lausanne, Switzerland, version 1.2, 2013.

[2] H. Nam, K.-H. Kim, J. Y. Kim, and H. Schulzrinne, "Towards QoE-aware Video Streaming using SDN," in *Proc. of the 2014 IEEE Global Communications Conference (GLOBECOM)*, Austin, TX, USA, Dec 2014, pp. 1317–1322.

[3] European FP7 Network of Excellence FLAMINGO Consortium (FLAMINGO NoE ICT-318488), "Deliverable D3.1 – Industrial Practice versus Academic State-of-the-Art," Filip De Turck and Olivier Festor, eds., 2013.

[4] Open Networking Foundation, "Software-Defined Networking: The New Norm for Networks," 2012, White Paper.

[5] M. Fiedler, T. Hossfeld, and P. Tran-Gia, "A Generic Quantitative Relationship Between Quality of Experience and Quality of Service," *IEEE Network*, vol. 24, no. 2, pp. 36–41, Mar 2010.

[6] P. Van Mieghem and F. A. Kuipers, "Concepts of Exact QoS Routing Algorithms," *IEEE/ACM Transactions on Networking*, vol. 12, no. 5, pp. 851–864, Oct 2004.

[7] The OpenDaylight Platform | OpenDaylight. Last access: September 17, 2015. [Online]. Available: http://www.opendaylight.org/

[8] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama, and S. Shenker, "Onix: A Distributed Control Platform for Large-scale Production Networks," in *Proc. of the 9th USENIX Conference on Operating Systems Design and Implementation (OSDI'10)*, Vancouver, Canada, Oct 2010, pp. 1–6.

[9] C. E. Rothenberg, M. R. Nascimento, M. R. Salvador, C. N. A. Corrêa, S. Cunha de Lucena, and R. Raszuk, "Revisiting Routing Control Platforms with the Eyes and Muscles of Software-defined Networking," in *Proc. of the First Workshop on Hot Topics in Software Defined Networks (HotSDN'12)*, Helsinki, Finland, Aug 2012, pp. 13–18.

[10] H. E. Egilmez, S. Civanlar, and A. M. Tekalp, "An Optimization Framework for QoS-Enabled Adaptive Video Streaming Over OpenFlow Networks," *IEEE Transactions on Multimedia*, vol. 15, no. 3, pp. 710–715, Apr 2013.

[11] H. E. Egilmez and A. M. Tekalp, "Distributed QoS Architectures for Multimedia Streaming Over Software Defined Networks," *IEEE Transactions on Multimedia*, vol. 16, no. 6, pp. 1597–1609, Oct 2014.

[12] S. Agarwal, M. Kodialam, and T. V. Lakshman, "Traffic Engineering in Software Defined Networks," in *Proc. of the 2013 IEEE Conference on Computer Communications (INFOCOM)*, Turin, Italy, Apr 2013, pp. 2211–2219.

[13] P. Georgopoulos, Y. Elkhatib, M. Broadbent, M. Mu, and N. Race, "Towards Network-wide QoE Fairness Using OpenFlow-assisted Adaptive Video Streaming," in *Proc. of the 2013 ACM SIGCOMM Workshop on Future Human-centric Multimedia Networking (FhMN)*, Hong Kong, China, Aug 2013, pp. 15–20.

[14] M. Jarschel, F. Wamser, T. Hohn, T. Zinner, and P. Tran-Gia, "SDN-Based Application-Aware Networking on the Example of YouTube Video Streaming," in *Proc. of the 2nd European Workshop on Software Defined Networks (EWSDN'13)*, Berlin, Germany, Oct 2013, pp. 87–92.

[15] T. Hoßfeld, D. Hausheer, F. V. Hecht, F. Lehrieder, S. Oechsner, I. Papafili, P. Racz, S. Soursos, D. Staehle, G. D. Stamoulis, P. Tran-Gia, and B. Stiller, "An Economic Traffic Management Approach to Enable the Triplewin for Users, ISPs, and Overlay Providers," in *Towards the Future Internet - A European Research Perspective*, 2009, pp. 24–34.

[16] O. Dobrijevic, A. J. Kassler, L. Skorin-Kapov, and M. Matijasevic, "Q-POINT: QoE-Driven Path Optimization Model for Multimedia Services," in *Wired/Wireless Internet Communications*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2014, vol. 8458, pp. 134–147.

[17] L. Skorin-Kapov and M. Matijasevic, "Modeling of a QoS Matching and Optimization Function for Multimedia Services in the NGN," in *Wired-Wireless Multimedia Networks and Services Management*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2009, vol. 5842, pp. 55–68.

[18] Open Networking Foundation, "OpenFlow Switch Specification: Version 1.3.4 (Protocol version 0x04)," 2014, Technical Specification.

[19] M. Dorigo and T. Stützle, *Ant Colony Optimization*. Cambridge, MA, USA: MIT Press, 2004.

[20] Z. Wang and J. Crowcroft, "Quality-of-service Routing for Supporting Multimedia Applications," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 7, pp. 1228–1234, Sep 1996.

[21] Mininet: An Instant Virtual Network on your Laptop (or other PC). Last access: September 17, 2015. [Online]. Available: http://mininet.org/

[22] N. Moura, B. Vianna, C. Albuquerque, V. Rebello, and C. Boeres, "MOS-Based Rate Adaption for VoIP Sources," in *Proc. of the 2007 IEEE International Conference on Communications (ICC)*, Glasgow, Scotland, Jun 2007, pp. 628–633.

[23] K. Yamagishi and T. Hayashi, "Parametric Packet-Layer Model for Monitoring Video Quality of IPTV Services," in *Proc. of the 2008 IEEE International Conference on Communications (ICC)*, Beijing, China, May 2008, pp. 110–114.

[24] D. Jurca, W. Kellerer, E. Steinbach, S. Khan, S. Thakolsri, and P. Frossard, "Joint Network and Rate Allocation for Video Streaming over Multiple Wireless Networks," in *Proc. of the 9th IEEE International Symposium on Multimedia (ISM 2007)*, Taichung, China, Dec 2007, pp. 229–236.