Global State-Dependent QoE based Routing

Hai Anh TRAN, Abdelhamid MELLOUK, Said HOCEINI, Brice AUGUSTIN
University of Paris-Est Creteil Val de Marne (UPEC)
Image, Signal and Intelligent Systems Lab-LiSSi Lab
Transport Infrastructure and Network Control Group - TINC
122 rue Paul Armangot, 94400 Vitry sur Seine, France
{hai-anh.tran, mellouk, hoceini, brice.augustin}@u-pec.fr

***Abstract*—For years, wireless network systems have been trying to satisfy end-users and support high quality multimedia applications such as Mobile TV, VoIP, etc. Combining wireless networks with multimedia content distribution needs efficient routing protocols. We develop in this paper a new routing protocol, namely DOQAR (Dynamic Optimized QoE Adaptive Routing), to improve the user perception and optimize the usage of network resources. In our end-to-end model, smartphone users connect to content servers in a wired network across a wireless access network. In order to evaluate the QoE, we use a Multi-Layer Perception-based (MLP) method. Experimental results show a significant performance against other traditional routing protocols.**

***Index Terms*—Quality of Experience (QoE), Quality of Service (QoS), network services, wireless network, routing system, Reinforcement Learning, autonomous system.**

I. Introduction

Nowadays, the Next Generation Networks (NGN) trend is the Fixed Mobile Internet Convergence (FMIC) in deploying Wireless Broadband Access (WBA) technologies and migrating the traditional telecom networks to the Internet Protocol (IP) technology. While NGN network experts are going to employ a common network layer protocol in core networks to accomplish the current network services, the access networks will use various technologies, such as WLAN, WPAN, Ethernet cable, DSL, 2G/3G, LTE, WiMAX, UWB, optical fiber, etc. to meet the diversified requirements of end-users [1]. Using a network environment with multiple operators and multiple networks, end-users expect to use a heterogeneous wired and wireless high-bandwidth ubiquitous network access and diversified services.

In fact, accessing the network from mobile devices (laptops, smartphones, or mobile phones) has now many choices for connectivity with the advent of Fourth Generation (4G) communication networks that is considered as a solution of an all-IP network layer. The notion of converged network is based on a model that combines a common network core having all network functionalities and different access networks. This combination develops a single network with various access technologies. With this increased choice, network service providers have to consider technical factors (i.e traditional QoS parameters) that influence the usefulness of the service.

While the known QoS concept designates a set of technical criteria to ensure network service, Quality of Experience

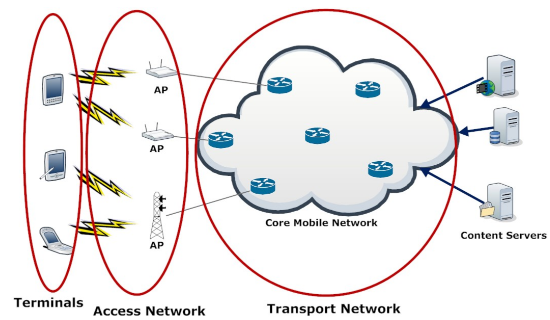


Fig. 1: End-to-end network system

(QoE) is a notion that represents the overall level of end-user satisfaction of a service. QoE is expressed by human feelings like “good”, “excellent”, “bad”, etc. A QoE-aware network system is a promising new notion in which service providers are aware of user perception and can consequently adapt to the dynamic environment to obtain acceptable and predictable QoE. In addition, the QoE impacts on the setting of internal parameters of the network.

Our work takes into account the end-to-end (e2e) QoE model (Fig. 1). The goal is to maintain the e2e quality between terminal users and servers through the network system. An e2e QoE system includes 3 components on which QoE has impact:

- *The user terminals and Content servers*: in a network system, the terminals represent end-user devices such as laptops, smartphones, etc. The QoE measurement can be realized in this part to give feedbacks into the network.
- *Access network*: end-users have to make a decision to choose which access network they use to connect to the system. Nowadays access network selection based on QoE is a new trend in NGN.
- *Transport network*: The core network may not be wired but may well be wireless by using “mobile routers”. In order to improve the network quality, QoE is considered as an efficient criteria to design routing methods in core networks.

According to wireless AP selection, various traditional approaches are based on connection quality between the AP and the end-user, but do not take into account the flow in the core network. Our approach is more general: the customer can choose between several access points, and it is based on which one offers the best QoE in the whole chain: terminals, access network and core network. In this paper, we focus on the routing mechanism in the core network to maintain the best QoE for end-users. Our approach is based on Q-Learning [2] which is one of the Reinforcement Learning (RL) algorithms. We evaluate the QoE at all nodes in the whole system.

For the same purpose, there are some related works: An overlay network for end-to-end QoE management is proposed by B. Vleeschauwer et al. [3]. G. Majd et al. [4] present a new QoE-aware adaptive mechanism to maximize the overall video quality on the client. M. Wijnants et al. [5] propose a QoE optimization platform that can reduce problems in the delivery path from service provider to users. Watkins and Daylan [6] propose a distributed end-to-end architecture for delivering streaming multimedia content. However, these approaches do not pay enough attention to user perception.

The paper is structured as follows: In section II, we focus on our proposed algorithm, the DOQAR algorithm, and explain how it can improve the end-user QoE. In section III, we present the QoE measurement testbed and the routing system experimental results. We end the paper with a conclusion and perspective.

II. Proposed Approach: DOQAR algorithm

Our algorithm is based on the Q-Learning algorithm. Based on the RL approach [2], Q-Learning [7] is an efficient algorithm to solve specific issues. Based on experience and feedbacks (rewards), an agent is able to learn control policies in the RL framework. Note that the underlying concept of RL is Markov Decision Process (MDP).

MDP model is illustrated by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$ with $\mathcal{S}$ is state's set, $\mathcal{A}$ is action's set and $\mathcal{P}(s'|s,a)$ is the transition model representing the probability of arrive to state s' after executing action a at state s. The reward obtained when the agent executes a at state s and enters s' is $\mathcal{R}(s,a,s')$. The quality of action a at state s is evaluated by Q-value $Q(s,a)$.

Solving an MDP is equivalent to finding an optimal policy $\pi : \mathcal{S} \to \mathcal{A}$, that maps states to actions such that the cumulative reward is maximized. In other words, we have to find the following Q-fixed points (Bellman equation):

$$Q^*(s,a) = r(s,a) + \gamma \sum_{s'} P(s'|s,a) max_{a'} Q^*(s',a') \quad (1)$$

where the first term on the right side represents the expected immediate reward of executing action a at s. The second term is the maximum expected future reward. γ is the discount factor. One of the most important breakthroughs in RL is an off-policy temporal difference (TD) control algorithm known as Q-learning, that directly approximates the optimal action-value function (Equation 1), independent of the policy being followed. The agent receives an immediate reward r from the environment after executing action a. Based on this reward and the long-term reward, the agent then updates the Q-values influencing future action selection. One-step Q-learning is defined as:

$$Q(s,a) = (1-\alpha)Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a')] \quad (2)$$

where α is the learning rate, which models the rate of updating Q-values.

In order to apply the Q-Learning algorithm into our routing system, we have mapped the RL model to our routing model in the context of learning a routing strategy. We consider each router along the routing path as a state and each link emerging from a router as an action to choose. The system's routing mechanism corresponds to the policy π.

Regarding QoE measurements (subsection II-C), we propose a MLP-based method that evaluates QoE at all nodes (routers) including the last one representing the wireless access point. In fact, measuring QoE anywhere in the system facilitates the application of Q-Learning to our model with the reward at any node.

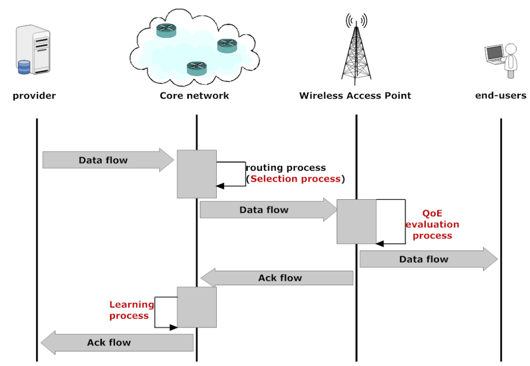


Fig. 2: DOQAR protocol

The proposed routing mechanism is illustrated in Fig. 2.

First step *- Data packet flow*: the provider sends a data packet to end-user. Each node in the routing path receives this data packet and then forwards it to the next one after evaluating the QoE value. The selection process is presented in detail in subsection II-A.

Second step *- At access point side*: When the data packet reaches the access point, it will be sent to end-users. The QoE evaluation process is realized to give a QoE feedback as an ACK message to the routing path that this data flow just followed. The QoE evaluation process is presented in subsection II-C.

Third step *- ACK message flow*: Every time a node receives an ACK message, it updates the Q-value of the link on which the ACK message was received. The update process is introduced below. The saved QoE result is attached into the ACK message, which is then forwarded to the previous neighbor.

In the next subsections, we present the selection and learning process. We introduce also our method to evaluate the QoE.

A. Selection process

In our approach, we take into account the the *exploration* and *exploitation* tradeoff. We cannot systematically *exploit* the link that has the maximum Q-value because each link must be evaluated many times to reliably estimate its expected reward. Therefore, we have chosen the softmax method using Boltzmann distribution [2]. With this softmax action selection rules, after receiving a packet, node x chooses its neighbor y_k among its n neighbors y_i ($i = 1..n$) with probability presented in Eq. 3:

$$p_{xy_k} = \frac{e^{\frac{Q_{xy_k}}{\tau}}}{\sum_{i=1}^{n} e^{\frac{Q_{xy_i}}{\tau}}} \quad (1 \leq k \leq n) \tag{3}$$

Where Q_{xy_i} represents Q-value of link xy_i and τ represents a temperature parameter of Boltzmann distribution. The link selection is all equi-probable with high temperature. On the contrary, low temperature generates a greater difference in selection probability for links that differ in their Q-values. In other words, the more we reduce temperature τ, the more we exploit the system. In that way, we reduce τ after each forwarding time as shown in Eq. 4:

$$\tau = \phi \times \tau \quad (0 < \phi < 1) \tag{4}$$

where ϕ is the weight parameter.

It is how we balance *exploration* and *exploitation*. Once the system has converged, we then exploit the system. The system convergence means all Q-values obtain the Q-fixed point (Eq. 1).

B. Learning process

In our model, each router has a routing table that indicates the Q-values of the outgoing links of this router. For example, node y has a routing table containing Q values: $Q_{yz_1}, Q_{yz_2}, Q_{yz_3}...Q_{yz_n}$ corresponding to n links from y to z_i with $i = 1..n$. In that way, we trivially construct the optimal routing path from a sequence of table look-up operations. Therefore, learning optimal routing strategy is equivalent to finding the Q-fixed points (Eq. 1).

As mentioned above, after receiving a data packet, the last node (representing the end-user) evaluates the QoE. It then sends back feedback in an ACK message to the backward routing path. Each router x in this routing path receives this message containing information: the QoE evaluation result (the MOS score) of the previous router (q_y) and the maximum value of Q-values ($\max Q_{yz}$) in the routing table of node y. Router x then updates the Q-value of the link connecting to y. Our update function based on the native Q-Learning (Eq 2) is defined in Eq. 5:

$$\underbrace{Q_{xy}}_{\text{new value}} = \underbrace{Q_{xy}}_{\text{old value}} + \alpha \left[\underbrace{\beta (q_y - q_x) + \gamma \max_i Q_{yz_i} - \underbrace{Q_{xy}}_{\text{old value}}}_{\text{new estimation}} \right] \tag{5}$$

Where Q_{xy} and Q_{yz_i} are Q-values of links xy and yz_i. q_x and q_y are results obtained (MOS score) by using QoE measurement method (presented at section II-C) at node x and y. α is the learning rate, which models the rate at which Q-values are updated. The two discount factors β and γ balance the value between future reward and immediate reward.

C. QoE evaluation process

The perceived quality of a multimedia stream is not easy to be evaluated because assessing perceived quality requires real people to evaluate it subjectively. In fact, subjective evaluations are very expensive and cannot be a part of an automatic process. Therefore, objective methods are considered as a solution to solve this problem. However they cannot replace subjective methods because their provided assessments do not perfectly correlate with human perception, so their use as quality measurement method is somehow limited.

In this paper, we propose a new technique that represents a hybrid between subjective and objective evaluation. Our QoE measurement method is based on the Neural Multi-Layer Perceptron (MLP) concept [8]. MLP is a feed-forward artificial neural network model mapping sets of input data onto a set of appropriate outputs. The formal neuron is a basic unit. It performs the weighted sum of its entries, and submits a nonlinear differentiable function.

For a formal neuron with n inputs, the neuron performs a weighted sum:

$$y = \sum_{i=1}^{n} \omega_i x_i \tag{6}$$

The output is then active with a linear function:

$$z = f(y) = f(\sum_{i=1}^{n} \omega_i x_i) \tag{7}$$

There are numerous activation functions, such as the gaussian function:

$$g(a) = exp(\frac{-a^2}{2}) \tag{8}$$

In our approach, we choose this latter as an activation function. In the following, we present the neuron network model we have used to evaluate the appropriate MOS score in our QoE measurement method. Our neural model includes:

- An input layer with five input cells x_i. This represents 5 QoS parameters: delay time, loss rate, conditional loss rate, bandwidth, RSSI (Receive Signal Strength Indicator)
- A hidden layer with six activation neurons y_j,
- An output layer with one neuron z representing the MOS score,
- 6×5 connections between the input layer and the hidden layer. Each one is weighted by v_{ji},
- 5×1 connections between the hidden layer and the output layer. Each one is weighted by ω_{kj}.

We chose empirically the neuron architecture with 6×5 connection between the input layer and the hidden layer, based on a series of experiments designed to select the most appropriate configuration.

We have implemented a testbed to provide the input for our estimation model. This testbed is presented in section III-A.

III. Experiments

A. Testbed for QoE measurement method

Training our estimator needs a real dataset of the impact of the network on the perceived video quality. To construct this dataset, we conducted an experiment consisting in selecting several people and asking them to score the perceived quality of a video using a MOS score. The testbed is composed by client-server architecture and a network emulator. The client is a VLC video client and the server is a VLC video streaming server [9]. The network emulator NetEm [10] is placed in the middle of the traffic between client and server and provides a way to reproduce a real network in a lab environment.

The experimental setup consists in forwarding video traffic between the server and the client. We use NetEm to change fixed delay, variable delay and loss values on the link to disturb and stress the video signal. The testbed is shown in Fig. 3.

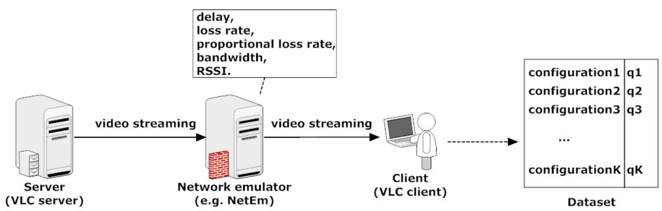


Fig. 3: Testbed for collecting the dataset

According to ITU-R [11], the length of the video should be at least 5sec. We choose the sintel video trailer [12]. This video is of 52 seconds duration, 1280 x 720 dimensions and 24 frames per second cadence and uses the H.264 codec. We chose this video because it alternates high and slow movements.

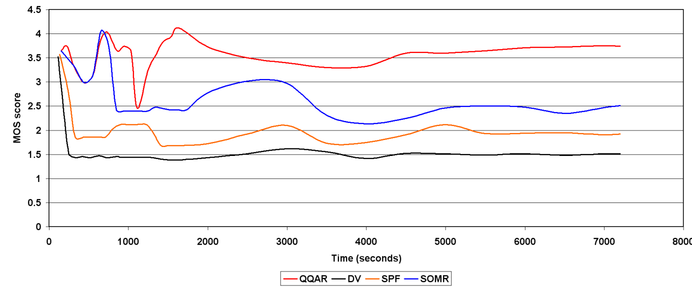
Experiments were conducted with fixed delay values of [25, 50, 75, and 100ms], variable delays of [0, 2, 4, 6, 8, 16 and 32ms], loss rate values of [0, 2, 4, 6, 10, 15, 20, 25 and 30%] and successive loss probability of [0, 30, 60 and 90%]. We chose these values to cover the maximum of QoE range. A number of viewers with a strong cinematic experience were chosen to watch the video and score them. We uses LCD monitors, since nowadays, a major part of monitors are LCD. We use the screens of 19" "LG flatron L194wt–SF" with 1440×900 resolution.

B. Routing system simulation results

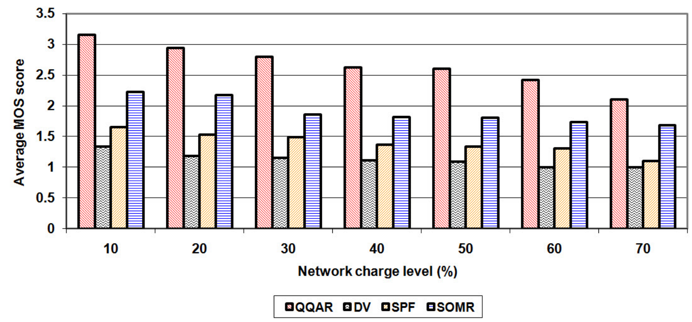
The Opnet simulator version 14.0 was chosen to implement our approach. Regarding network topology, we have implemented an irregular network with 3 separated areas including 38 mobile routers, each area being connected to each other by one wireless link. We dynamically change the parameters of the network system every of 200 seconds.

To validate our results, we compare our approach with three types of algorithm:

- *Distance-Vector (DV) algorithm* [13]. In this algorithm, to maintain the optimal route, all routers store the address of the next router in the routing table so that the number of hops to reach the destination is minimal.



(a) User perception under low traffic load



(b) User perception under different levels of traffic load

Fig. 4: User perception

- *Link-State algorithm: SPF (Short Path First)* [14]. In this algorithm, the routers use hello messages to establish relations with each other. The Dijkstra's algorithm is used to determine the shortest path to each network known in the LSDB (Link-State Database).
- *Standard Optimal QoS Multi-Path Routing (SOMR) algorithm* [15] where routing is based on finding the K-Best Optimal Paths and uses a composite function to optimize delay and link cost criteria simultaneously.

Fig. 4 illustrates the result of an average MOS score of the four algorithms (non-loaded network in 4a and different load levels in 4b). In the heavy load scenario, we have generated a traffic that stresses the network. The load level represents the rate of number of loaded links and total number of links:

$$\text{level} = \frac{n_s}{N} \tag{9}$$

where n_s is the number of loaded links and N is the total number of links in the system.

In Fig. 4a, in the first 25 minutes all algorithms fluctuate very much. That is explained by the execution of an initialization process. In other words, this fluctuation is explained by the exploration phase of each algorithm. In these first 1500 seconds, the MOS score of DOQAR varies between 2.5 and 4, SOMR between 2.4 and 4.1, SPF between 1.7 and 3.6, DV between 1.4 and 3.5. After the first 1500 seconds, the protocols gradually become stable. DOQAR varies between 3.4 and 3.7. DV and SPF are quite stable with average results respectively 1.4 and 1.9. SOMR varies much more but the maximum value (obtained in period from the 2100th to the 2300th second) is still lower than DOQAR. Recall the qualitative values of MOS: 1 - Bad, 2 - Poor, 3 - Fair, 4 - Good and 5 - Excellent.

Fig. 4b gives us the average of these four algorithms in different load levels formulated in (9): from 10% to 70%. We can see that the more the system is loaded, the more the average score decreases. However, at any charge level, the average MOS score of DOQAR is better than the three other algorithms. With a load level of 10%, the MOS score of DOQAR is higher than 3 (which represents a fair quality). Regarding the three other classical protocols, the SOMR obtains a maximum value of 2.2 in load level of 10%.

DOQAR gives a better e2e QoE perception than the three other algorithms under different levels of traffic load. So with our approach, although the network environment changes, we can maintain a better QoE. Thus, in response to changes in network dynamics , DOQAR is able to adapt its decisions rapidly.

Our experiment also consists in the evaluation of overheads caused by these protocols. We take into account the overhead control that is determined by the proportion of control packets compared to the total number of packets emitted. To monitor this overhead value, we have varied the node number in adding more routers into network system. The observed node numbers are [38, 50, 60, 70, 80]. The obtained results are showed in Fig. 5.

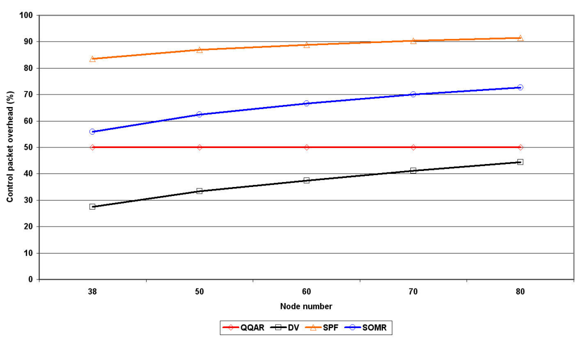


Fig. 5: Control overhead

We can see that the control overhead of our approach is constant (50%). That is explained by the equal number of control packets number and data packets in DOQAR. Each generated data packet leads to an acknowledgement packet generated by the destination node. The control packet rates of DV, SPF and SOMR are respectively 0.03, 0.4 and 0.1 (packet/second). We can see that in Fig. 5. The SPF algorithm has the highest control packet rate (0.4 packet/second) with multiple types of packets, so it has the highest value of overhead. With a control packet rate value of just 0.03, the DV algorithm has the smallest overhead value. We can see also that the higher the number of nodes is, the higher the overhead is. Therefore, our approach is more scalable than the three others with a stable overhead.

IV. Conclusion

Routing mechanisms in the end-to-end system is an important issue to be considered in future multimedia networks. QoE can impact on three components of an e2e network system: Terminals, Access Network and Core Network. We focus on the routing issue in a Core mobile network with mobile routers and wireless links. In this scenario, we introduce a QoE aware routing protocol, named DOQAR. Experimental results show that our routing protocol gives significant QoE evaluation improvements over traditional routing approaches. In the future, we plan to integrate the selection of the best access network based on user QoE feedbacks in order to treat the whole chain.

References

[1] J. Zhang and N. Ansari, "On assuring end-to-end QoE in next generation networks: challenges and a possible solution," *Communications Magazine, IEEE*, vol. 49, no. 7, pp. 185–191, 2011.

[2] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *IEEE transactions on neural networks*, vol. 9, 1998.

[3] Bart De Vleeschauwer et al., "End-to-end QoE Optimization Through Overlay Network Deployment," *International Conference on Information Networking*, 2008.

[4] G. Majd, V. Cesar, and K. Adlen, "An Adaptive Mechanism For Multipath Video Streaming Over Video Distribution Network (VDN)," *First International Conference on Advances in Multimedia*, 2009.

[5] M. Wijnants *et al.*, "Optimizing user quality of experience through overlay routing, bandwith management and dynamic transcoding," *International Journal of Adaptive, Resilient and Automatic Systems*, vol. 1, no. 2, p. 64, 2010.

[6] J. Famaey *et al.*, "Dynamic QoE optimisation for streaming content in large-scale future networks," in *Integrated Network Management-Workshops, 2009. IM'09. IFIP/IEEE International Symposium on*. IEEE, 2009, pp. 128–134.

[7] Watkins and Daylan, "Technical note: Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, 1992.

[8] M. Poulton, "Multi-layer perceptrons and back-propagation learning," *Handbook of Geophysical Exploration: Seismic Exploration*, vol. 30, pp. 27–53, 2001.

[9] Videolan. [Online]. Available: http://www.videolan.org/

[10] S. Hemminger, "Network emulation with netem," in *Linux Conf Au*, April 2005.

[11] "Recommendation 500-10: Methodology for the subjective assessment of the quality of television pictures," ITU-R Rec. BT.500, 2000.

[12] Sintel video trailer. [Online]. Available: http://www.sintel.org/

[13] G. Malkin, "RFC 2453: Rip version 2," *Request for Comments*, vol. 13, no. 04, p. 2010, 1998.

[14] J. Moy, *OSPF: anatomy of an Internet routing protocol*. Addison-Wesley Professional, 1998.

[15] R. Banner and A. Orda, "Multipath routing algorithms for congestion minimization," *IEEE/ACM Transactions on Networking (TON)*, vol. 15, no. 2, pp. 413–424, 2007.