Autonomic Quality of Experience Management of Software-Defined Networks

Madalena Pereira da Silva[*], Mario Antonio Ribeiro Dantas[*,**]
Alexandre Leopoldo Gonçalves[*]

[*]Department of Engineering and Knowledge Management
[**]Department of Computer Science
Federal University of Santa Catarina
madalena.silva@posgrad.ufsc.br, m.a.r.dantas@ufsc.br, a.l.goncalves@ufsc.br

Abstract. Quality of Experience (QoE) refers to the subjective evaluation of the user's perceptions and expectancies during the use of a service. In order to understand it regarding a given service and how to instrumentally stimulate it, one must know and model the different factors and parameters related with technological and human agents. As important as that, is to conceive tools for managing QoE centered in the user, which adopting a top-down approach, are able to adapt the resources in the subjacent network infrastructure according to the user's experience. This study presents an Autonomic Semantic Engine for QoE Management in Software-Defined Network (SDN) architecture. The Engine management cycle together with the SDN control application contribute to the management of QoE with fine granularity, since it minimizes manual settings and allows the MAPE-K control loops to detect QoE violations and make decisions for providing the service according to the user's experience.

1 Introduction

Quality of Experience (QoE) refers to the subjective evaluation of the user perceptions and expectancies during the use of a service. In order to understand it regarding a given service and how to instrumentally stimulate it, one must know and model the different factors and parameters related with technological and human agents. As important as that, is to conceive tools for managing QoE centered in the user, which adopting a top-down approach (from the user to the network), are able to adapt the resources in the subjacent network infrastructure according to the user's experience.

During the past years, the term "Quality of Experience" has generated a lot of documents indexed in scientific databases. The researches have been motivated in order to provide and deliver services that meet and exceed the user's expectations. However, most of them approach the QoE as an extension of QoS (Quality of Service), only prioritizing the technological aspects and thus neglecting other dimensions that interfere with QoE. In addition, proposals are implemented in the infrastructure of current networks architecture, which makes impossible the dynamic configuration of the network in order to meet different demands. Given these findings, the question is: how to make QoE management able to satisfy different needs of the user?

A possible alternative is to use intelligent mechanisms capable of learning the user's experience in the use of a service, and provide information to the control applications can perform adjustments on network devices to prevent degradations in the service.

Motivated by those challenges, this research presents the main elements to model QoE and how they may be included in a QoE Management Architecture in Software-Defined Network (SDN). The SDN architecture is enriched with autonomic and semantic features to provide the QoE management with fine granularity, minimize the degree of human interference for the restoration of QoE and allow the autonomic element to take optimization decisions of QoE from a knowledge base (KB).

Contributions proposed are (i) a taxonomy of dimensions impacting QoE, based on an interdisciplinary approach and (ii) an Autonomic Semantic Engine that learns the user's experience using a service and provides information so that the control plan may perform adjustments in the network elements, in case of QoE degradation.

A set of dimensions, from the proposed approach, is analyzed considering a multimedia scenario, where an IPTV service is provided, based on the user's experience, adopting the flexibility of the SDN paradigm.

This paper is organized as follows: Section 2 addresses the impacting dimensions in QoE. Section 3 proposes the modeling of QoE. Section 4 presents a functional architecture, incorporating the engine and the elements of the QoE modeling. Section 5 describes a testbed and experimental results. Section 6 summarizes the related work. Section 6 presents our conclusions and future works.

2 Survey of impacting dimensions on quality of experience

Organized in three stages, this part of the research aims to evidence the dimensions impacting on the QoE.

2.1 Capture of terms from the opinion survey

Aiming to know which aspects are considered as important ones to contemplate or to overcome the expectancies in a service provision, an opinion survey with 167 contacts from Facebook social network was carried out. An analysis on the 124 answers received allowed a classification of terms in four categories: user, context, content and technology.

2.2 Matching terms capture the QoE concepts

Results from the opinion survey evidenced that researches involving user experience must go beyond the technological aspects and the knowledge of profiles. The user experience must be evaluated in a broader way, including the cognitive, aesthetic and hedonic needs. This evaluation is only possible if worked out in an interdisciplinary way. In order to study QoE in the interdisciplinary context, concepts of QoE were extracted from literature that: (i) consider the subjective and objective human factors when formulating their definitions and (ii) have some term equivalent with the factors captured in the opinion survey. QoE concepts are described in (ITU-T, 2008), Fiedler et al. (2010) and Wu et al. (2009). From those definitions four dimensions of QoE influence were extracted: user, content, context and technology. Those dimensions are equivalent to the ones collected in the opinion survey.

2.3 Research of papers about QoE

Aiming to investigate if the dimensions impacting QoE (user, content, context and technology) have been explored, a systematic review of literature was carried out. Table 1 summarizes the research, highlighting the approach, the types of applications and the conceptual bases which support the proposals.

Paper	Approach	Application	Theorical Basis	QoE
Li-yuan et al. (2006)	Disciplinary	Video streaming in pervasive computing	Context aware computing, Fuzzy	QoS, Context, user profile, subjective human factors
Wechsung et al. (2012), Möller et al. (2009)	Interdisciplinary	Multimodal man-machine interation	HMI, Interface Design	QoS, user, context, system, objective human factors
Moor et al. (2008)	Interdisciplinary	Quality in cell phone development	HMI, Interface Design, TIC, Engineering, Sociology	Quality of application, user context, objective and subjective human factors
Wu et al. (2009)	Interdisciplinary	Distributed Interactive Multimedia Environments	Psychology, Cognitive Science, Sociology, TIC	Quality of application, user context, objective and subjective human factors
Skorin-Kapov and Varela (2012)	Interdisciplinary	Video Stream, bidirectional VOIP	Psychology, Sociology, TIC	Quality of application and network, resource, user context, objective and subjective human factors
Laghari et al. (2010)	Interdisciplinary	Service delivery in distributed environments	Psychology, Cognitive Science, Sociology	User, service, business, subjective human factors
Chen and Kotz (2000)	Interdisciplinary	Service delivery and provisioning in distributed environments	Psychology, Cognitive Science, Sociology, HMI, TIC, Interface Design	QoS of application and network, service, businnes, context, objective and subjective human factors

TAB. 1 – *Research summarizattion: multidimensional QoE.*

The papers presented indicate that the context dimension has been little explored. Neither proposal addresses the issue of content considering the semantic meaning from the user's point of view. Besides, the proposals are designed to the current Internet architecture and thus limited in providing the same QoS for different users. Moreover, in the event of possible network degradation, the links may not be re-configured and this reflects negatively in the user's experience.

3 Proposed QoE modeling

Through research it is clear that the proposals involving user experience must go beyond technology and knowledge of one's profile. User experience should be treated more broadly, including cognitive, aesthetic and hedonic needs. This assessment is only possible if treated in an interdisciplinary way.

3.1 Taxonomy of the quality of experience

Once the survey of impacting dimensions on quality of experience has been presented, it was possible to map four dimensions (Human, content, context and quality). Fig. 1 evidences the factors that influence the user satisfaction when using a service. The same ones are described below.

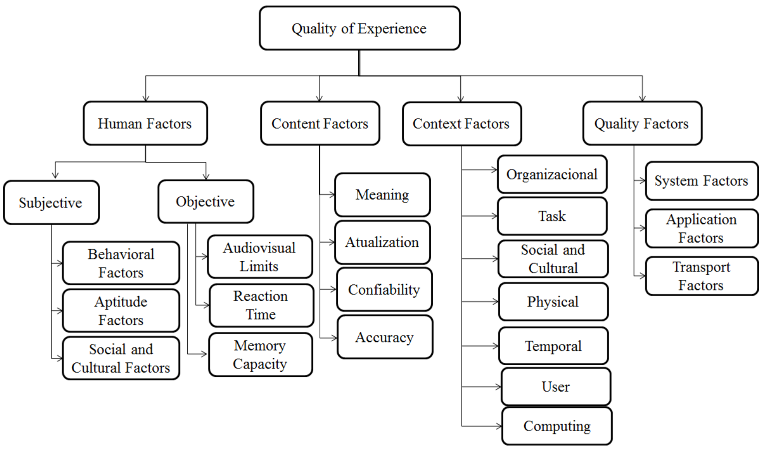

FIG. 1 – *Taxonomy of the quality of experience dimensions.*

Subjective human factors. Those may be acquired by opinion surveys, interviews. The purpose is to know which aspects are considered to be important in order to fulfill or surpass the expectancies in the provision of a service. For capture and analysis of each aspect in a real environment, psychology theories, HMI (Human-Machine Interface), Interface Design may be used. The choice about which theory to adopt depends on the nature of service, environment, context and which variables are important to consider in the study. Those theories have been used in the following works (Laghari et al., 2011) and (Wu et al., 2009).

Objective human factors. They are used to predict the performance of the human task, such as memory capacity, audio-visual human limits, reaction time of the system. They may be obtained by controlled or uncontrolled surveys, using invasive or noninvasive methods. They are supported by the conceptual bases of Cognitive Science, Cognitive Psychology and Mental Models. Authors Wu et al. (2009) and Laghari et al. (2010) have those disciplines as theoretical bases to found their proposals.

Context factors. Factors related with context have been pointed out in a cluster of researches. For Bellavista et al. (2013) and Chen and Kotz (2000) context is contemplated in four dimensions: computing, physical, time, user. However, there are other dimensions of context which must be considered Moor et al. (2008), such as organizational context, task, social and cultural, because those allow to model knowledge in a wider approach and to help the reasoner to infer more precisely.

According to Nazário et al. (2012), from context information, the services may be adapted, adjusting to characteristics of the current environment and from the user profile. In other words, it is possible to provide more optimized and customized services, in order to offer a greater quality of experience to users. The use of context may also minimize the consumption of resources such as energy, processing and communication, delivering more precise and dynamic services.

For capture the information of temporal context and location (Chen and Kotz, 2000), sensors of the equipment may be used, such as GPS, 3G, etc. In Bellavista et al. (2013) is described how the computation context (processing power, memory, battery, bandwidth) can be captured and processed. Information of social and cultural context (Cardone et al., 2012) may be obtained with the user agreement, from his social networks; microphone to analyze the social situation (chatting, transit, office); accelerometer to discriminate the physical activity (running, walking, driving). The task context may be obtained according to the type of service being solicited by the user. The organizational context and other information may be obtained by directly asking the user.

Content factors. knowledge media is concerned both with word language (semantic communication), as well as the technical language of signals (syntactic communication), Pavanati et al. (2010). In modelling proposed QoE, knowledge media is being approached with those two visions, however, both objects of study are studied in different dimensions in Fig. 1. In other words, the syntactic communication is studied in the "Quality of Service" dimension, with all parameters which affect its QoS being considered. And the semantic communication is studied in the dimension "Quality of Content".

The content factors which impact QoE are dependent on the representation, value, meaning of the information for the user. Semiotics (Costa et al., 2011) is one of the areas that deals with word meanings, the value that the receptor gives to the message according to its semantics. This value may be influenced by context and by the human being subjectivity.

The meaning that the user attributes to a concept may be mapped into ontologies. Using ontologies it is possible to associate several synonyms to a same concept, according to the user semantics, as well as to disambiguate words (D' Agostini and Fileto, 2009) associating the concept to a context. Example: associate the term "Saint Paul" to the context (football, city, Saint).

Quality of service factors. It is related with quality parameters of the application (e.g. codec, bitrate resolution); of network performance (e.g. delay, jitter, packet loss, bandwidth). Once all QoS requirements of the application have been treated, the transport system receives and transport the data in a channel ready to warrant QoS so that the information is delivered for the user with a good performance.

The parameters of network QoS may be captured by monitoring and network control software. Mapping parameters of network QoS, with the level of user satisfaction, whose service was evaluated in MOS scale, for example, it can be possible, during execution time, to dynamically adjust QoS level in the commuting elements in order to not degrade the user QoE. This flexibility is facilitated with the use of SDN, allowing the dynamical allocation of resources.

For subjective evaluation of audio and video, MOS (ITU-T, 2003) is one of the mostly used. To evaluate other services, not related with multimedia applications, another evaluation method with cognitive approach, based in emotions, has been mentioned in literature (Berg and Ulbricht, 2013).

We consider that the most effective way of measuring QoE is from the local where the service is being consumed, in other words, in the user's terminal. Thus, QoE knowledge needs to be represented and transported to a QoE management system.

3.2 Knowledge representation

To monitor the QoE, a management system receives information from service sessions from several terminals, in different contexts. It is necessary to standardize concepts, allowing interoperability and machine reasoning. One possibility is by using ontologies. Another advantage of using ontologies is their reuse. Thus, an already validated ontology was searched in the literature for this purpose Fallon and O'Sullivan (2012). However, in order to represent the dimensions of the proposed taxonomy (Fig. 1), the ontology (Fallon and O'Sullivan, 2012) was extended, according to Fig. 2. Concepts in blue were kept as in the original ontology while concepts in grey were created to reflect the proposed taxonomy.

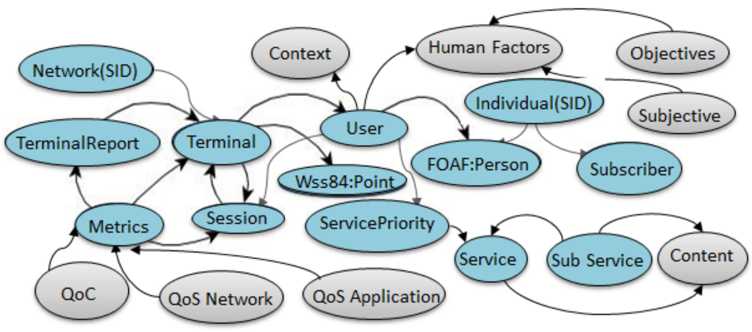

FIG. 2 – *Part of the ontology for representing QoE knowledge. Source: adapted from Fallon and O'Sullivan (2012)*

Table 2 summarizes the purpose of each concept, with reuse of ontologies (Fallon and O'Sullivan (2012), Brickley (2007) and Fallon and Huang (2011)).

Concept	Role
TerminalReport	Represents the report generated by the user terminal, contains other information from the ID, TIME STAMP and REPORT TYPE concepts.
Terminal	Represents a user's terminal, contains other information from the ID, ADDRESS, TYPE, MANUFACTURE, SERIAL NUMBER concepts.
Context	Represents the user's context and terminal. Other concepts, illustrated in Taxonomy in Fig.1, may be specialized therefrom.
User	Represents the user, contains other information from the ID, ADDRESS concepts.
HumanFactors	Represents the objective and subjective user parameters, other concepts, illustrated in Taxonomy in Fig. 1, may be specialized from these.
FOAF: Person	Represents information about a person as described in the FOAF ontology.
Wsg84:Point	Represents the terminal location information through the LATITUDE, LONGITUDE and ALTITUDE concepts.
Metrics	Represents the metric in the user's terminal, contains other information from the START TIME, DURATION and RESULT CODE concepts.
QoEMOS	Represents the metric subjective MOS.
QoENetwork	Represents objective metrics of performance parameters, contains other information from the THROUGHPUT, LATENCY JITTER and PACKET LOSS concepts.
QoSApplication	Represents the application's parameters combined in the session, contains other information from the RESOLUTION, CODEC and GOP concepts, in the case of multimedia applications.
Session	Represents the session service / sub active service, contains other information from the ID and TYPE concepts.
Service	Represents the service in use, such as VoIP , IPTV, WEB.
SubService	Represents the services accessible from another service. Example, under the IPTV servisse, the following services are available: Video on Demand (VoD), Music on Demand (MoV), Game on Demand (GOV), Pay Per View (PPV), Interactive TV (iTV), eHealt - provided by a third party.
Content	Represents the type of content used during the session service / sub service.

TAB. 2 – *Key concepts of QoE ontology from the user's terminal. Source: adapted from Fallon and O'Sullivan (2012).*

A database of policies which allows to compare provided quality vs. expected quality is needed so that the management system can detect QoE degradations. Ontology was created for this so that network managers may define different policies for the services. Fig. 3 illustrates a simplification of it and how the concepts relate with the QoE ontology (Fig. 2).

Table 3 summarizes the purpose of each ontology concept of QoS policies.

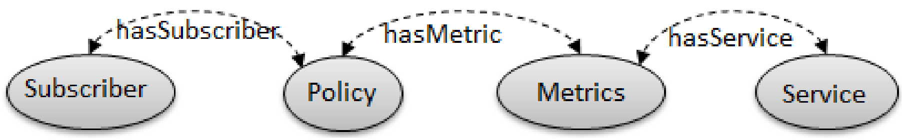

FIG. 3 – *Part of the ontology for representing QoS policies.*

Concept	Role
Subscriber	Identifies the subscriber of the service
Policy	Is responsible for associating a QoS policy for each contracted service
Metrics	Represents each QoS metric associated with the service
Service	Represents the contracted service

TAB. 3 – *Key concepts of policy ontology.*

3.3 Negotiating, monitoring and transport metrics of QoS/QoE protocol

By using a protocol, the terminal may report information from the session to the system of QoE management, including the service, application, user and content of each session.

As the protocol is generic, it may be used in any terminal and support semantics. GSQR (Generic Service Quality Reporting Protocol for Terminals), Fallon and Huang (2011), is the most indicated protocol. GSQR supports metrics for services such as IPTV, Web, conversation and conference, as well as subservices associated with each one of the services.

Reports are hierarchically organized in XML, with semantic annotations referencing concepts of ontology, with records by service, subservices and by terminal. They may be generated in the beginning, during and at the end of sessions.

4 Autonomic architecture driven semantics for the QoE management using SDN

In order to provide a QoE management architecture, we preserved and included new features in the framework proposed by the ONF [19] and the MAPE-K Control Loop proposed by IBM (Keller, 2005). The architecture consists of three levels, as shown in Fig. 4.

4.1 Management plan

This plan contains the business applications of the organization, which offer network services. Any application of this level communicates with the standard SDN controller or with some application module of the control plan, using the Northbound API. The main contribution, in this plan, consists in the Autonomic Semantic Engine. The engine is constituted by QoE Ontology; Policy Ontology; Knowledge Base and MAPE-K Control Loop.

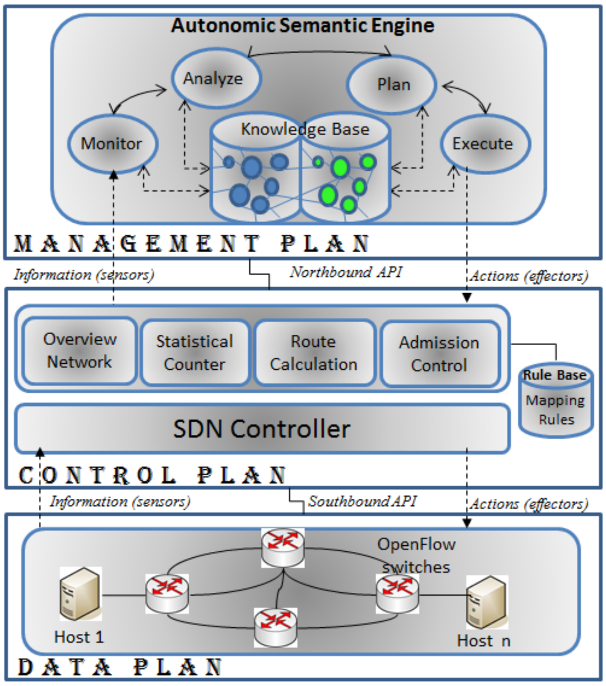

FIG. 4 – *Functional architecture for QoE management.*

4.1.1 QoE ontology, Policy ontology and knowledge base

The QoE ontology, described in section 3, it is used to unify concepts, easing the knowledge about the user's experience when using the service and allowing inferences. The policy ontology, described in section 3, it is used to formalize policies per user and per service. The Knowledge Base (KB) holds information on the managed resources (network and user terminals); policies and policy adaptation actions. All the actions taken by control loop are made based on policies and other existing information in the KB.

4.1.2 MAPE-K Control Loop.

Consists of the functions (Monitor, Analyze, Plan, Execute and knowledge), creating a sequence of actions necessary to the autonomic element to make decisions from the KB.

Monitor module. Monitors the managed elements (network resources and user's terminal) according to predefined policies.

In order to monitor network resources, probes or application modules about the SDN controller can be used in the control plan. Fig. 4 proposes the implementation of the "Network Overview" and "Statistics Counter" Applications. The control plan application modules provide information to the management plan using a Northbound API.

In order to monitor the quality of the user terminal sessions, a capture module is required. This module performs the process of Extract, Transform and Load (ETL) information in the reports from the user's terminals. The information in the reports is structured in XML format, referencing the concepts of QoE ontology. Transport of data from the terminal was made using a protocol adapted from Fallon and Huang (2011).

Analysis module. This module works with the information in the KB and is composed by semantic rules used to analyze and verify QoE degradations. The module compares the expected value of the metric with values (minimum and maximum limits) of policies defined by the administrator (policies ontology). When degradations are detected, an event from the "Analysis Module" is triggered for a "Plan Module" that queries its policies database about actions to be performed according to the metric.

Plan module. This module performs the planning policies of adaptations of actions to correct the degradation of the QoE. Table 4, adapted from Bari et al. (2013) illustrates some examples of policy adaptations that may be applied for optimizing QoE.

QoS Metrics	Policies Adaptation Actions
Dropping packets	Change queue configuration. Forward flow through alternative route.
Throughput	Change limiting rates of flows saturating bandwith. Forward flow through alternative route.
Latency	Change queue configuration of the switch. Plan the transmission of flows through a less congested route with proper delay
Jitter	Forward flow through a less congested route

TAB. 4 – *Examples of policies adaptations for optimizing QoE. Source: adapted from Wu et al. (2009).*

When a policy change is needed, the high level rules are converted for control rules and and persisted in Rule Base of control plan. Planning considers the QoS policy as in the subscriber's service contract. Policies for subscribers and services are described through Service Level Agreement (SLA). The objectives of the SLA include the prioritization of service e QoS guaranteed network.

Execute module. In order to apply the actions, this module communicates with the control plan using a Northbound API. At this time, the high-level rules are persisted as control rules, on the "Rule Base", of the control plan. After the control plan verifies the available resources, the actions are performed in the commuters interfaces (queues, routes, flow limiters), using a Southbound API.

4.2 Control plan

It is composed by the SDN controller, which is able to communicate with commuters from the data plan, via Southbound APIs, using, for instance, the OpenFlow protocol. In order to allow autonomic settings for optimizing QoE, this plan needs some control applications in addition to a standard SDN controller, such as those proposed in Bari et al. (2013). Note that the contribution of this paper consists not in implementing those modules, but in the mechanisms described in the management plan, which in turn provide information for the control layer modules to perform the adaptation of network policies. Control applications are described hereinafter according to Bari et al. (2013).

Admission control application. Receives solicitations "Execute Module" from the management plan, analyzes the network resources (i.e. queues, entries in the flow table, bandwidth) and accepts or rejects the requisitions according to the availability of resources.

Overview network application. Knows the state up/dow of OpenFlow switches and their doors by listening to the asynchronous messages (Packet-In, Flow-Removed, Port-Status) exchanged between the OpenFlow controller and switch.

Route calculation applicatio. Discovers the routes available and select the best route based on the overall state of the network information (topology, performance metrics), persisted in the Rule Base.

Statistics Counter. Collects information on network performance metrics (for instance, bandwidth, delay, jitter, packet discard) and generates statistics for flow, port and link.

Rule base. Are used by the management plan for mapping the high level network policies into control rules. The controller and the control applications (for instance, routing) use those rules for calculating the entries in the flow table of each OpenFlow switch.

4.3 Data plan

Composed by network devices with some protocol enabled. OpenFlow (Foundation, 2011) was the first open protocol to provide standard APIs for managing the data plan in the commuting structure by the controller. The controller uses the OpenFlow API to discover the network topology; view the status of devices and active flows; adds and removes forwarding rules in the flow tables of network switches, monitor and consult statistics for flows, queues and ports.

OpenFlow provides support for monitoring and controlling QoS. The packages pertaining to a flow may be queued in a particular queue of an output interface. Controllers may consult configuration parameters and the statistic of queues. Switches may rewrite the IP ToS field in the IP header. Support for notification of explicit congestion is also inserted in OpenFlow.

ONF (Foundation, a) created the OF-Config (Foundation, b) for supporting the configuration of several characteristics in an OpenFlow switch, including the configuring the minimum and maximum transmission rates of a queue in an OpenFlow switch. An OpenFlow controller may also read those rates from a switch. The most recent additions regarding QoS in OpenFlow are the meter tables and meter bands, which may be used for limiting the transmission rate of an output interface.

5 Testbed and experimental results

This section presents a possible scenario (Fig. 5) and the experiment performed for demonstrating how the Autonomic Semantic Engine, using a substrate SDN, may be applied for detecting and fixing problems in the case of QoE degradation.

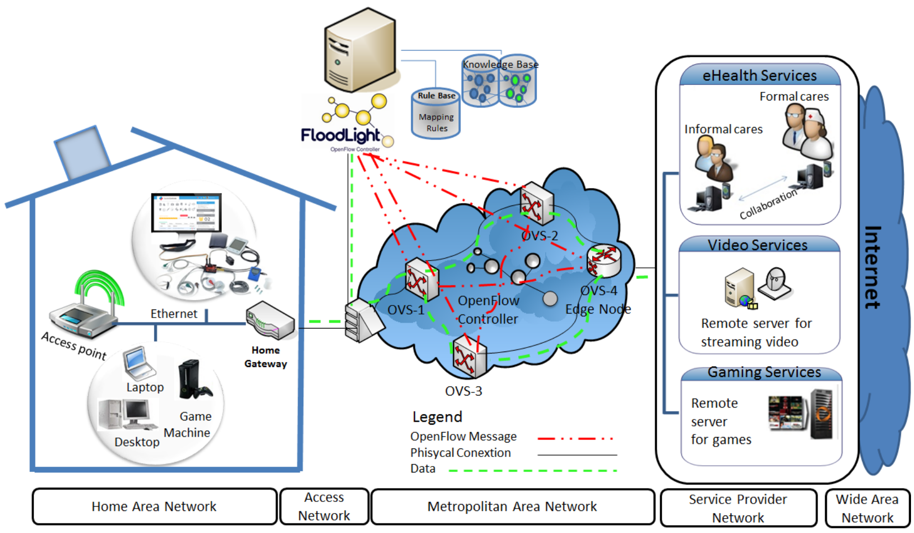

FIG. 5 – *Topology of the network used in the experimental scenario.*

5.1 Scenario description

In the experiment the QoE of flows in active sessions of three users is analyzed, which use different applications, in a HAN (Home Area Network) consuming three sub-services from the IPTV provider. In opportune moments background traffic is generated in order to ascertain the behavior of the proposed mechanism. Experiments were performed in laboratory.

User Alice is a remotely assisted patient that through the IPTV service will use the eHealth sub-service to send monitoring images and vital signs information to a remote unity, and receive information back from the remote unit. User Bob, through IPTV will use the sub-service of GoD (Game on Demand). Another user will use the VoD (Video on Demand) service.

5.2 Resources and equipment used on testbed

In order to simulate HAN three hosts were used, representing the user's machines, a gateway and biomedical signal sensors. For biomedical sensors configuration, was used e- Health

Sensor platform (Cooking-hacks), published in our previous research Nazário et al. (2014). To simulate the remote patient monitoring cameras were used.

Hosts have client applications with interfaces for accessing the services and a module for capturing the impact dimensions of QoE installed on them.

For simulating the service provider network and testing the proposed approach, a server machine was used. This machine was configured with two objectives: - to act as the Autonomic Network Management Application and Application Server.

The application server (VoD, GoD, eHealth and other services) was configured in Ubuntu Linux 13.04, 64bits. In the server some videos were made available, as well as games and files so that users could watch and play by the network and download data. Files to download and Health e-Learning were made available to simulate the e-Health service. The data from biomedical sensors and patient images are sent to the same server.

5.3 Management plan

Implemented the Autonomic Semantic Engine with the QoE ontology and policies ontology, controol loop and Knowledge Base (KW).

5.3.1 Ontologies, SWRL rules and KB

The ontologies and semantics rules were built in Protégé 4 (Research). The KB is populated with information from managed resources; policies by service and user; policy adaptations actions. It is powered and/or used by the controol loop functions.

5.3.2 MAPE-K control loop

Each module of the "control loop" was implemented in Java as a REST application. These applications communicate with the control plan, using REST API.

Monitor. Captures information from managed resources (user's terminals, network elements). In order to capture information from the user's terminal, a module was prototyped in Java, using the SAWSDL framework (W3C) for Extract, Transform and Load (ETL) information in the reports from the user's terminals. Information in the reports is structured in XML format, referencing the concepts of QoE ontology. Information from the terminal reports, including values of QoE and QoS performance parameters are transported in XML. The module was configured so that from time to time the terminal sends information about QoE and QoS parameters from reports to the "Monitor Module" of the service provider network. When the reports arrive in the QoE management system, they pass through the ETL processing, and information is persisted in the KB, as instances in OWL (Web Ontology Language).

Analysis. Uses semantic rules to infer new facts and detect degradations of QoE. The rule base was built in SWRL. This application makes use of OWL-API everytime it is necessary to work with instances in OWL, and SWRL rules. For each performance metric (i.e. delay, jiiter, disposal), it is created a semantic rule that compares whether the metric monitored in the user terminal is being provided in accordance with the thresholds defined in the policies ontology.

If there is violation of the SLA, an event from the "Analysis Module" is triggered for a "Plan Module".

Plan. Determines policy ajustment actions so as to optimize QoE according to the violation of the metric. The high level rules of the management plan are translated as control rules. To map the QoS of high-level network policies (i.e., queue switch interface; minimum bandwidth) for control rules, it is necessary to specify a minimum set of parameters with JSON encoding.

Execute. Communicates with the control plan so that the QoE optimization actions are performed in the network elements. The control rules, specified in JSON are persisted on the "Rule Base" of the control plan. Depending on the availability of resources, actions are performed and the state of the network is updated in the control loop KB.

5.4 Control plan

The Floodlight controller was installed and used as an OpenFlow controller. It was adopted because it has a group of modules and applications that, together with the OpenFlow API and the OF-Config protocol, allow the visualization of the network topology, status of devices, changing the forwarding tables and verifying active flows, among other functionalities.

For this scenario, the modules Topology Management; Static Flow Entry Pusher and Counter Store of Floodlight were used for, respectively, verifying the network topology, forwarding the flows, installing and removing flows from a given switch and generating statistics. For monitoring performance parameters of the transport network, Linux tools and the Counter Store module of Floodlight were used. The output of the monitor's data was used as input for the controller.

5.5 Data plan

Contains the managed resources, i.e., user terminals and switches openflow mesh used for the transport of data. In order to emulate the Metropolitan Area Network, four OVS (Open Virtual Switch) were used, installed in an Ubuntu 13.04-64bits operating system. Each OVS runs in a separate physical machine, interlinked using GRE tunnels (Generic Routing Encapsulation). The tc command of Linux was used for configuring the maximum capacity and the delay of each link.

5.6 Experimental Results

Two experiments were performed. In both experiments, the Autonomic Semantic Engine was activated. The first experiment consists in testing the throughput adaptation policy with alternative route. Two routes were created (route 1: S1-S2-S4 and route 2: S1-S3-S4) for sending the flows through alternative paths. Initially all flows follow route 1. QoS warranties of 5Mbps, 3Mbps and 2Mbps, were established respectively for eHealth, VoD and GoD. eHealth flows run from Host 4 (application server) to Host 1 (Alice's machine). VoD flows from Host 4 to Host 2 (Bob's machine). GoD flows from Host 4 to Host 3 (another user's machine). In order to induce QoE degradation, at 10 seconds, the generation of background flows was programmed, transmitted from Host 4 to Host 3, with a transmission rate of 4Mbps.

All flows are transported by route 1. With that transmission rate, bandwidth was saturated (limited in 10Mpbs to send HAN) and the services started to suffer degradation from 11 until 15 seconds. The "Module Analyze" of Autonomic Semantic Engine compares the measured values with the defined values detected violation of the throughput metric. An event was triggered for "Module Plan".

The "Module Plan", queried its policies database, examine the kind of action to be performed and translates the high level of rule to control rule. The "Module Execute" is triggered and using the Northbound API, communicated with the control plan. The high level rule was persisted in the Rule Base and the "Admission control" application was invoked. As this application has an overall view of the network, it verified the availability of alternate rules and the module "Static Flow Entry Pusher" of the controller was triggered for changing the forwarding table on the switches and to forward the badly behaving flows through route 2 (S1-S3-S4). Around 15 seconds throughput was reestablished and the network status updated in KB, as Fig. 6 illustrates.

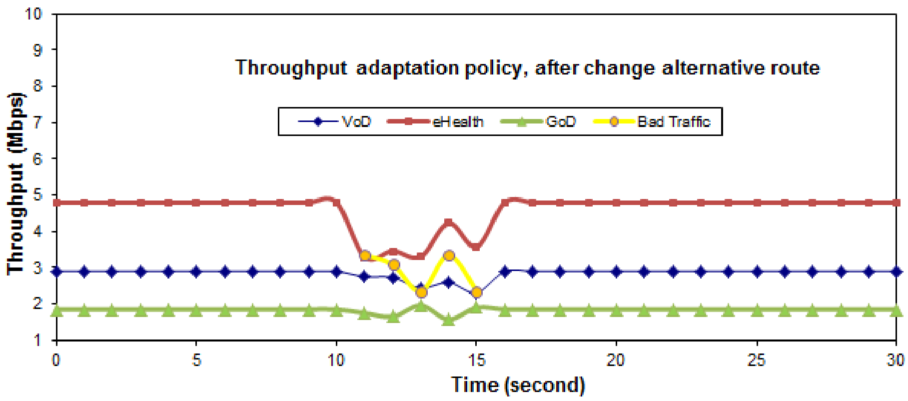

FIG. 6 – *Autonomic adaptation of the throughput metric, with alternate route.*

The second experiment consisted in testing the throughput adaptation policy, without alternate route. In this case, only route 1 was kept for testing the system behavior, packing best effort traffic and traffic with QoS warranties, with bandwidth saturation. QoS warranties of 4Mbps, 2Mbps and 1Mbps were established, respectively for eHealth, VoD and GoD. Until 30 seconds the throughput was kept in all sessions. From this moment, Bob's terminal started to heavy games downloads from the server and at 41 seconds, Alice's terminal started to update the eHealth system from the server. Those new sessions compromised the bandwidth of active flows, until the problem was detected and fixed by the Autonomic Semantic Engine and the control applications, as Fig. 7 illustrates.

The "Module Analyze" of "Autonomic Semantic Engine" detected violation in the values of the throughput metric and generated events for the "Module Plan". As the flows are "best effort" and in this experiment there is no alternate route for deviating them, the "Module Plan"

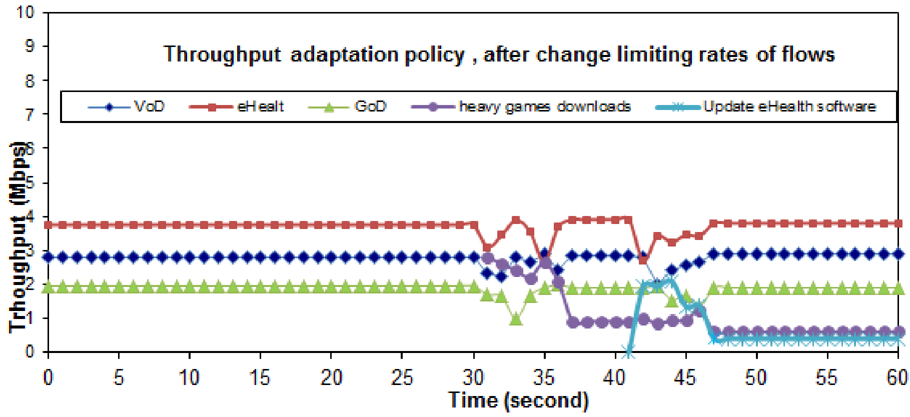

FIG. 7 – *Autonomic adaptation of the throughput metric, without alternative route.*

searched its policies database and found that the action to be performed consisted in changing the rate of limiting flows, because they were saturating the bandwidth.

The "Module Plan" translates the high level of rule to control rule. The "Module Execute" is triggered and using the Northbound API, communicated with the control plan. The high level rule was persisted in the Rule Base. The admission control application verified the available resources and, using the OpenFlow protocol, through the Southbound API, the limiting rate for flows (updating antivirus software and e-Health) were changed on the switches interfaces. Around 37 and 47 seconds throughput was reestablished and the network status updated in KB, as Fig. 7 illustrates.

6 Related work

The papers presented in Table 6 provided information to set out this proposal. However, our proposal differs from the others as it deals with QoE as a multidimensional construct; it is based on an interdisciplinary approach; it adds semantic and autonomic features for QoE management in networks projected for Future Internet.

An investigation into related papers allowed the mapping of the existing proposals. The search filtered papers where QoE was studied in an interdisciplinary and/or multidimensional way or where QoE was designed for the Future Internet using SDN. Those proposals using a formal basis for the representation of knowledge were also considered; and whether they had autonomic features and were designed for networks of the Future Internet. Table 5 summarizes and shows where our proposal stands out.

By providing a consistent ontological base and a data transport protocol, the paper described in Fallon and O'Sullivan (2012) is the most relevant one to our study, even not being designed for the Internet of the Future. We extended the ontology proposed by that study in order to reflect the taxonomy in Fig. 1, and we implemented our own data transport protocol based on XML.

Paper	Interdisciplinary	Multidimensional	Management		Autonomic	Semantic	SDN
			QoE	QoS			
Fallon and O'Sullivan (2011)	X	X	X	X	X	X	
Bari et al. (2013)				X	X		X
Kim et al. (2012)				X	X	X	X
Latre et al. (2008)			X	X	X	X	
Proposal	X	X	X	X	X	X	X

TAB. 5 – *Summarization of the related work.*

Another significant study proposed in Bari et al. (2013) allowed us to define the control applications needed for communication of the Autonomic Semantic Engine with the control layer. However, we have implemented our own control applications based on Floodlight modules, API REST and API OpenFlow.

Although we did not use or extend new functionalities, we considered the studies described in Kim et al. (2012) and Latre et al. (2008) as related, and a rich source of research. Additionally, study Kim et al. (2012) provides a comprehensive view of the use of autonomic computing with a semantic approach to SDN. And study Latre et al. (2008) provides a Generic Knowledge Base for optimizing Media Access Networks.

7 Conclusions and further research

In this work the modeling QoE based in the cause-effect relationship between QoE and QoS was presented. The taxonomy about the impact dimensions of QoE was proposed, as well as ways to capture information from those dimensions, based on related work. The functionality of the model received an autonomic semantic engine for Quality of Experience Management in Software-Defined Networks.

The Autonomic Semantic Engine was proposed to make policy adaptation decisions from the KB. It is able to recognize the user's experience using the system and to map information from different dimensions (user, context, content, QoS) in knowledge bases. Other information such as the status of network performance and user evaluations about the service, are captured and stored in the KB. With all this information it is possible to allocate resources, define entries in forwarding tables of flow commuters, as well as to discover new facts, such as, not allowing the user QoE to be affected by degradation in the QoS parameters of the network.

The Semantic Engine Autonomic management cycle together with the SDN control application contribute to the management of QoE with fine granularity, since it minimizes manual settings and allows the MAPE-K control loops to detect QoE violations and make decisions for providing the service according to the user's experience.

Experimental results demonstrate that the autonomic and semantic approach, when incorporated in a SDN architecture, allows that the managing system of the provider can learn about the user's experience, based in a context and provides information for the controlling applications to take actions for restoring QoE if degradations of performance metrics are evidenced. In the experiments it was observed that the time spent between QoE detection and the adaptation of network policies is acceptable not interfering in the quality perceived by the user.

The research work continues with the implementation of new functionalities in the application of QoE/QoS management MAPE-K control loops for testing adaptations of policies using other performance metrics. Besides, the combination of other dimensions of QoE, proposed in the Taxonomy (Fig. 1) must be combined and experimented in order to contemplate, to the maximum, the user's expectancies when using a service.

References

Bari, M., D. R., S. Chowdhury, R. Ahmed, and R. Boutaba (2013). Policycop: an autonomic qos policy enforcement framework for software defined networks. *Future Networks and Services (SDN4FNS), 2013 IEEE SDN for*, 1–7.

Bellavista, P., A. Corradi, M. Fanelli, and L. Foschini (2013). A survey of context data distribution for mobile ubiquitous systems. *ACM Computing Surveys (CSUR) 44*, 1–49.

Berg, C. H. and V. Ulbricht (2013). Emotional usability testing for human computer interface. *Educaonline 7*, 84–100.

Brickley, D. (2007). Basic rdf geo vocabulary. Technical report, W3C Semantic Web Interest Group (http://www.w3.org/2003/01/geo/).

Cardone, G., A. Corradi, L. Foschini, and R. Montanari (2012). Socio-technical awareness to support recommendation and efficient delivery of ims-enabled mobile services. *Communications Magazine, IEEE 82-90*, 50.

Chen, G. and D. Kotz (2000). A survey of context-aware mobile computing research. Technical report.

Cooking-hacks. e-health sensor platform v2.0 for arduino and raspberry pi [biometric/medical applications]. Tutorial, (http://www.cookinghacks.com/documentation/tutorials/ehealth-biometric-sensor-platformarduino-raspberry-pi-medical).

Costa, A. R. C., M. M. Bortolato, and R. P. L. d. Sousa (2011). Semiotic study of graphic-interactive digital interface picasso head. *semiotic studies (USP) 7*, 114–125.

D' Agostini, C. S. and R. Fileto (2009). Capturing user's preferences and intentions in a semantic search system. *SEKE - International Conference on Software Engineering& Knowledge Engineering*, 587–591.

Fallon, L. and Y. Huang (2011). Gsqr: A generic service quality reporting protocol for terminals. *Integrated Network Management (IM 2011) IFIP/IEEE International Symposium on*, 1122–1128.

Fallon, L. and D. O'Sullivan (2012). Using a semantic knowledge base for communication service quality management in home area networks. *Network Operations and Management Symposium (NOMS 2012)*, 43–51.

Fiedler, M., S. Karlskrona, T. Hossfeld, and P. Tran-Gia (2010). A generic quantitative relationship between quality of experience and quality of service. *IEEE Network Magazine 24*, 36–41.

Foundation, O. N. Technical report, ONF Open Networking Foundation.

Foundation, O. N. Management and configuration protocol (of-config). Technical report, ONF Open Networking Foundation.

Foundation, O. N. (2011). Openflow switch specification. Technical report, ONF Open Networking Foundation. Version 1.1.0 Implemented (Wire Protocol 0x02).

ITU-T, G. (2008). G1080: Quality of experience requirements for iptv services. Recommendation itu-t g.1080, International Telecommunication Union.

ITU-T, P. (2003). Mos - recommendation p.800.1, mean opinion score terminology. Mos - recommendation p.800.1, International Telecommunication Union. MOS - Recommendation P.800.1, Mean opinion score terminology.

Keller, A. (2005). Towards autonomic networking middleware. *2nd International Workshop on Next Generation Networking Middleware*.

Kim, S.-S., S.-s. Seo, J.-M. Kang, and J. Hong (2012). Autonomic fault management based on cognitive control loops. *Network Operations and Management Symposium (NOMS)*, 1104–1110.

Laghari, K. u. R., I. G. B. Yahya, and N. Crespi (2010). Towards a service delivery based on customer experience ontology: shift from service to experience. *MACE, Modelling Autonomic Communication Environments 6473*, 51–61.

Laghari, K. u. R., I. G. B. Yahya, and N. Crespi (2011). Qoe aware service delivery in distributed environment. *Advanced Information Networking and Applications (WAINA), IEEE Workshops of International Conference on*, 837–842.

Latre, S., P. Simoens, B. De Vleeschauwer, W. Van de Meerssche, F. De Turck, B. Dhoedt, P. Demeester, S. Van Den Berghe, and E. de Lumley (2008). Design for a generic knowledge base for autonomic qoe optimization in multimedia access networks. *Network Operations and Management Symposium Workshops (NOMS)*, 335–342.

Li-yuan, L., Z. Wen-an, and S. Jun-de (2006). The research of quality of experience evaluation method in pervasive computing environment. *Pervasive Computing and Applications, 2006 1st International Symposium on 178-182*, 178–182.

Möller, S., K. P. Engelbrecht, C. Kühnel, I. Wechsung, , and B. Weiss (2009). A taxonomy of quality of service and quality of experience of multimodal human-machine interation. *Quality of Multimedia Experience. QoMEx 2009. International Workshop on 7-12*, 7–12.

Moor, K. d., L. d. Marez, T. Deryckere, W. Joseph, and L. Martens (2008). Bridging troubled water: quality of experience in a mobile media context. *Terena Networking Conference*.

Nazário, D., I. Tromel, M. A. R. Dantas, and J. L. Todesco (2014). Toward assessing quality of context parameters in a ubiquitous assisted environment. *Computers and Communication (ISCC), 2014 IEEE Symposium on*, 1–6.

Nazário, D. C., M. A. R. Dantas, and J. L. Todesco (2012). Taxonomy of publications on context quality. *Sustainable Business International Journal 20*, 1–28.

Pavanati, I., R. P. L. d. Sousa, and O. d. R. Neves Jr (2010). Representation, reality and knowledge in the digital-electronic media. *Texting digital (UFSC) 6*, 2–18.

Research, B. I. The protégé ontology editor and knowledge acquisition system. Technical report, Biomedical Informatics Research (BMIR). (http://protege.stanford.edu/).

Skorin-Kapov, L. and M. Varela (2012). A multi-dimensional view of qoe: the arcu model. *MIPRO, Proceedings of the 35th International Convention*, 662–666.

W3C. Semantic annotations for wsdl and xml schema. W3c recommendation, W3C (http://www.w3.org/TR/sawsdl/). W3C Recommendation 28 August 2007.

Wechsung, I., K. P. Engelbrecht, C. Kühnel, S. Möller, and B. Weiss (2012). Measuring the quality of service and quality of experience of multimodal human machine interaction. *Multimodal User Interfaces 6*, 73–85.

Wu, W., A. Arefin, R. Rivas, K. Nahrstedt, R. M. Sheppard, and Z. Yang (2009). Quality of experience in distributed interactive multimedia environments: toward a theoretical framework. *MM '09 Proceedings of the 17th ACM international conference on Multimedia*, 481–490.

Résumé

La Qualité d'Expérience (QoE) se réfère à l'évaluation subjective des perceptions et des attentes de l'utilisateur lors de l'utilisation d'un service. Afin de comprendre ce qui concerne un service donné et comment le stimuler instrumentalement, il faut connaître et modéliser les différents facteurs et paramètres liés avec des agents technologiques et humaines. Ce qui est tout aussi important, c'est concevoir des outils pour gérer QoE centrée sur l'utilisateur, qu'en adoptant une approche top-down, sont en mesure d'adapter les ressources dans l'infrastructure du réseau sous-jacente selon l'expérience de l'utilisateur. Cette étude présente un moteur sémantique autonome pour la gestion de la QoE dans l'architecture de réseau définie par logiciel (SDN). Le cycle de gestion du moteur avec le logiciel de contrôle SDN contribuent à la gestion de la QoE avec une granularité fine, car les réglages manuels sont minimisés et les boucles de contrôle MAPE-K peuvent détecter les violations de la QoE et prendre des décisions pour fournir le service en fonction de l'expérience de l'utilisateur.