Over the Top Content Streaming Adaptive System- Implementation and Validation

Serban Georgica Obreja, Radu Iorga, Eugen Borcoci,
Cristian Cernat, Marius Vochin
University POLITEHNICA of Bucharest
Bucharest, Romania
Emails: serban@radio.pub.ro,
radu.iorga@elcom.pub.ro
eugen.borcoci@elcom.pub.ro
cristian.cernat@elcom.pub.ro
marius.vochin@elcom.pub.ro

Jordi Mongay Batalla
National Institute of Telecommunications
Warsaw, Poland
email: jordim@interfree.it

Daniel Negru, Joachim Bruneau-Queyreix
LaBRI Lab, University of Bordeaux
Bordeaux, France
Emails: daniel.negru@labri.fr
jbruneauqueyreix@labri.fr

***Abstract* — Adaptive content streaming is an efficient and cheap solution to achieve a good Quality of Service and Experience for media streaming, having an Over- the –Top light architecture, i.e., working on top of the current IP technologies. The adaptation in the system considered here consists in content server initial optimized selection based on multi-criteria algorithm and then in-session media adaptation (using dynamic adaptive streaming) and/or server switching. In this paper, a first set of results on the implementation and functional validation of the system are presented. It focuses on presenting some of the functional validation scenarios and the tests results of the server selection component.**

Keywords — Content delivery; Dynamic Adaptive Streaming over HTTP; Quality of Service; Quality of Experience; Monitoring, Server and Path selection.

I. INTRODUCTION

The light Over-the-Top (OTT) architectures are recently developed for media/content delivery over the current public Internet. These are more simple and cheaper in comparison to complex solutions involving the network resources management and control architecture, like Content Oriented Networking [1].

This paper considers an OTT-style content streaming system, proposed, designed and implemented by the European *DISEDAN* Chist-Era project [2], (*service and user-based **DI**stributed **SE**lection of content streaming source and **D**ual **A**daptatio**N**,* 2014-2015). The business actors involved are: *Service Provider (SP)* delivering the content services to the users and possibly owing and managing the transportation network; *End Users (EU)* that consumes the content; a *Content Provider (CP)* could exist, owning some *Content Servers (CS)*. In DISEDAN light architecture, it is assumed that CSs are also owned by the SP.

The DISEDAN solution novelty [2][9]-[12], consists in: (1) *two-step server selection mechanism* (at SP and at EU) using algorithms that consider context- and content-awareness and (2) *dual adaptation mechanism* consisting of *media adaptation* and *content source adaptation* (by *streaming server switching*) when the quality observed by the user suffers degradation during the media session.

For in-session media adaptation, the *Dynamic Adaptive Streaming over Hypertext Transfer Protocol - HTTP (DASH)* technology has been selected. The DASH is a recent multimedia streaming standard, to deliver high quality multimedia content over the Internet, by using conventional HTTP Web servers [3-6]. It minimizes server processing power and is video codec agnostic. A DASH client continuously selects the highest possible video representation quality that ensures smooth play-out, in the current downloading conditions. This selection is performed on-the-fly, during video play-out, from a pre-defined discrete set of available video rates and with a pre-defined granularity (according to video segmentation).

This presents the DISEDAN testbed where the system components and samples of validation scenarios have been implemented. This paper study is focused on phase 1 of functioning, i.e., initial server selection. The specific DASH topics are treated in other studies [12].

The paper structure is the following. Section II is a short overview of related work. Section III outlines the overall DISEDAN architecture and main design decisions. Section IV contains the paper main contributions, focused on DISEDAN system testing and validation performed on a real life testbed. Section V contains conclusions and future work outline.

II. RELATED WORK

Assuring a "good" Quality of Experience (QoE) based on QoS control at server and transport levels is an important feature of the real time media related services. Apart from resource provisioning method, which supposes a complex management infrastructure at network level, adaptive methods are recently considered as good solutions [3][4].

Media flow adaptation is used in recent standards [5][6], as a significant technique to improve the QoE. *Content server switching* during session (assuming that replica servers are available) can additionally be decided if the media adaptation no longer produces good results. To these two, the DISEDAN solution adds an initial server selection phase (based on cooperation between SP and EU) integrating all three in a single solution. The initial server selection can

 ISBN: 978-1-61208-455-8

be based on optimization algorithms like *Multi-Criteria Decision Algorithms (MCDA).* In [8][9], several scenarios are proposed, analyzed and evaluated, considering the availability of different static and/or dynamic input parameters. Therefore several control plane design decisions are possible [10], different in complexity/performance. The dynamic capabilities for the initial CS selection and then for adaptation decisions depends essentially on the power of the DISEDAN monitoring system [10]. The system combines in a novel solution the DASH functionalities with additional monitoring in order to finally realize the dual adaptation.

III. SUMMARY ON DISEDAN SYSTEM ARCHITECTURE AND DESIGN DECISIONS

This section provides a short description of the DISEDAN architecture and main design decisions. More complete description is given in [10][11][12].

The connectivity between CSs and EU Terminals (EUT) is assured by traditional *Internet Services Providers (ISP) / Network Providers (NP*) - operators. The ISP/NPs do not enter explicitly in the business relationships set considered by DISEDAN, neither in the management architecture. Service Level Agreements (SLAs) might be agreed between SP and ISPs/NPs, related to connectivity services offered by the latter to SP; however such SLAs are not directly visible at DISEDAN system level.

The system is flexible; it can work either in OTT style, or over a managed connectivity service offered by the network. An implicit assumption is that network environment is the traditional TCP/IP. No reservation for connectivity resources, neither connectivity services differentiation at network level are supposed (but they are not forbidden). The SP does not commit to offer strong QoS guarantees for the streaming services provided to EUs, therefore no SLA relationships between EUs and SPs management entities is mandatory. However, it is assumed that a Media Description Server exists, managed by SP, to which EUT will directly interact.

The media streaming actions are transport-independent. The EUT works as a standalone application; no mandatory modifications applied to the conventional SP; however, SP should provide some basic information to EUT, to help it in making initial server selection (and optionally to help in-session CS switching). The decision about dual adaptation (media flow adaptation and/or CS switching) is taken mainly locally at EUT, thus assuring user independency and avoiding complex SP-EUT signaling during the session. Several CSs exist, known by SP; so, the SP and/or EUs can operate servers' selection and/or switching. DISEDAN does not treat how to solve failures, except attempts to do media flow DASH adaptation or CS switching.

The work [11] has defined all requirements coming for EU, SP and derived the general and specific system requirements and some assumptions and constrains. The architecture (Figure 1) has been determined by such requirements. Details on that are included in [11]. The functional blocks correspond respectively to SP, EUT and CS. Note that only relevant to DISEDAN blocks are shown in the picture.

The Service Provider (SP) includes in its Control Plane:

MPD File generator – dynamically generates Media Presentation Description (MPD) XML file, containing media segments information (video resolution, bit rates, etc.), ranked list of recommended CSs and, optionally - current CSs state information and network state (if applicable).

Selection algorithm – runs Step 1 of server selection process. It exploits *MCDA* [7][8] to rank the CSs and media representations.

Monitoring module – collects information from CSs and processes it to estimate the current state of each CS.

The End User Terminal (EUT) includes the modules:

Data Plane: DASH (access and application) – parses the MD file received from SP and handles the download of media segments from CS; *Media Player* – playbacks the downloaded media segments. The standard ISO/IEC 23009-1, "Information technology -- Dynamic adaptive streaming over HTTP (DASH)" [5], defines the DASH-Metrics client reference model, composed of *DASH access client (DAC)*, followed by the *DASH-enabled application (DAE)* and *Media Output* (MO) module. The DAC issues HTTP requests (for DASH data structures), and receives HTTP request responses.

Control Plane: Content Source Selection and Adaptation engine – implements the dual adaptation mechanism; *Selection algorithm* – performs the Step 2 of server selection process. It can also exploit MCDA, or other algorithms to select the best CS from those recommended by SP; *Monitoring module* – monitors changing (local) network and server conditions.

The CS entity includes the modules:

Data Plane: Streaming module – sends media segments requested by End Users; *Monitoring module* – monitors CS performance metrics (CPU utilization, network interfaces utilization, etc.).

Figure 1 shows the main functional steps: (1) EUT issues to SP a media file request. (2) SP analyzes the status of the CSs and runs the CS selection algorithm. (3) SP returns to EUT an ordered list of candidates CS (SP proposal, embedded in a MD- xml) file. (4) The EUT finally selects the CS, by running its own algorithm. (5) EUT starts asking segments from the selected CS. During media session, the EUT makes quality and context measurements. Continuous media flow DASH adaptation is applied, or, (6) CS switching is decided. From the EU point of view, the steps 1-2-3 composed the so-called Phase1 and steps 4-5-6 composed the so-called the Phase 2.

During the receipt of consecutive chunks, the user's application can automatically change the rate of the content stream (based on DASH measurements, which are out of scope in this paper) and/or also can switch to another CS.

The design decisions are taken for the Control Plane, details on this being given in [10].

 ISBN: 978-1-61208-455-8

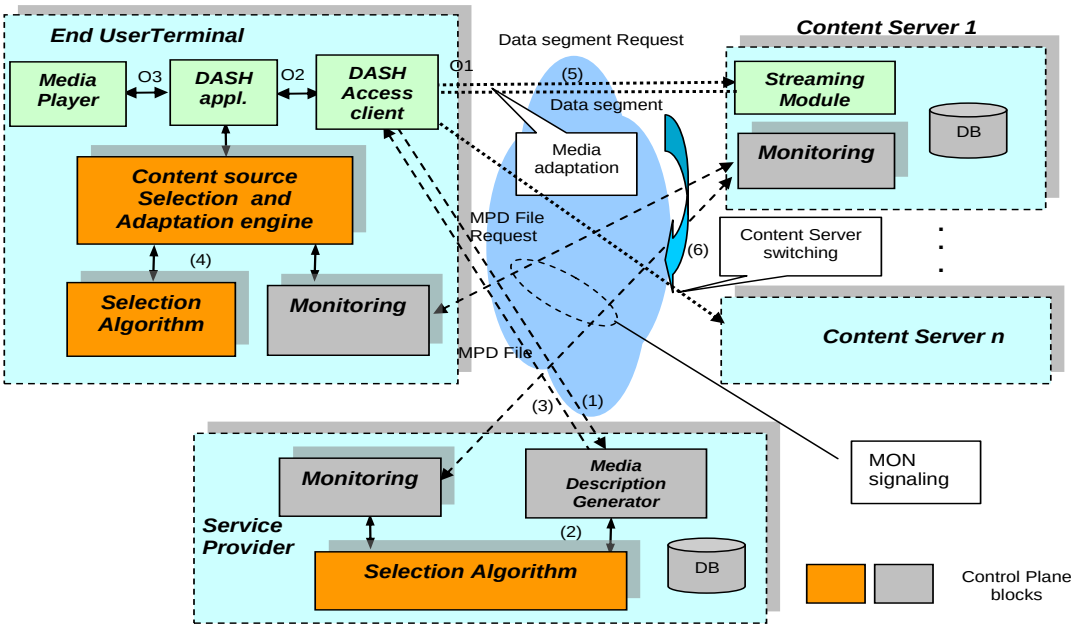

Figure 1. DISEDAN general architecture

IV. DISEDAN SYSTEM TESTING AND VALIDATION

This section is dedicated to present the test configuration and validation scenarios. The overall goal is to validate the content server selection phase in different server and network load conditions. Two sample scenarios, out of the complete set performed, are given as examples in this paper.

A. *Testbed configuration*

An experimental testbed has been built for DISEDAN functionalities validation (Figure 2). The system comprises three independent IP network domains, equipped with several core and edge/border routers (Linux based). No QoS technologies are active in these networks. Several DISEDAN entities are connected to this network: SP, EU, and CS through some access networks. Note that the presence of the access networks in the overall system is not essential, given the OTT-style of work for DISEDAN.

B. *Basic signaling test and validation*

The sequence of steps for functional validation scenario is illustrated in Figure 2 as a set of actions:

1. The EUT requests a streaming service from SP. The request contains the ID of the media service requested.

2. The SP gets from its local database the identity of the servers hosting the requested content. Using its monitoring module, the SP interrogates the monitoring agents located on the CSs about the CS state. The main monitoring parameters collected by SP are: CS processor load, CS free memory (normalized at the total memory), number of streaming processes on CS, total bandwidth on the network interface.

3. After receiving the monitoring parameters, the SP selects, using the MCDA, the best servers from the list of servers hosting the requested resource.

4. An ordered list of selected servers is returned to EUT. The best server from SP's point of view is on the first position on the list.

5-6. The EUT performs a second selection step. The *Round Trip Time (RTT)* and the distance in terms of *hop count* are measured from EUT to the CSs in the list recommended by SP. The communication on fifth step is performed only with the SP's selected servers not with all CSs. Based on this metric the best server is selected by EUT.

Final step: the EUT requests the service from the selected CS; the latter will start streaming packets towards the EUT.

All the functional steps described above have been validated by capturing and analyzing the messages exchanged between EUT, CS, SP.

C. *Validation of the server selection algorithm in different servers load conditions*

This test illustrates how the CS server's selection is influenced by the load of the content servers. The same setup as the one illustrated in Figure 2 was used.

In the first phase, the terminal requests from Service Provider a new service. The request follows the steps of the selection procedures, as described in the previous section, till it is accepted by the SP and the terminal gets the movie stream from the selected server. The selection of the best CS servers is done by the MCDA algorithm running on SP. It selects the best server based on the list with the network distance between the terminal and the content servers and the lists with the CPU load on content servers, the number of sessions on CSs, the load on the serving network interface on CSs, and the occupied memory on CSs. The values of these parameters are first normalized and then passed to the MCDA algorithm organized into a matrix, MCDA matrix, as it is described in [8][9].

 ISBN: 978-1-61208-455-8

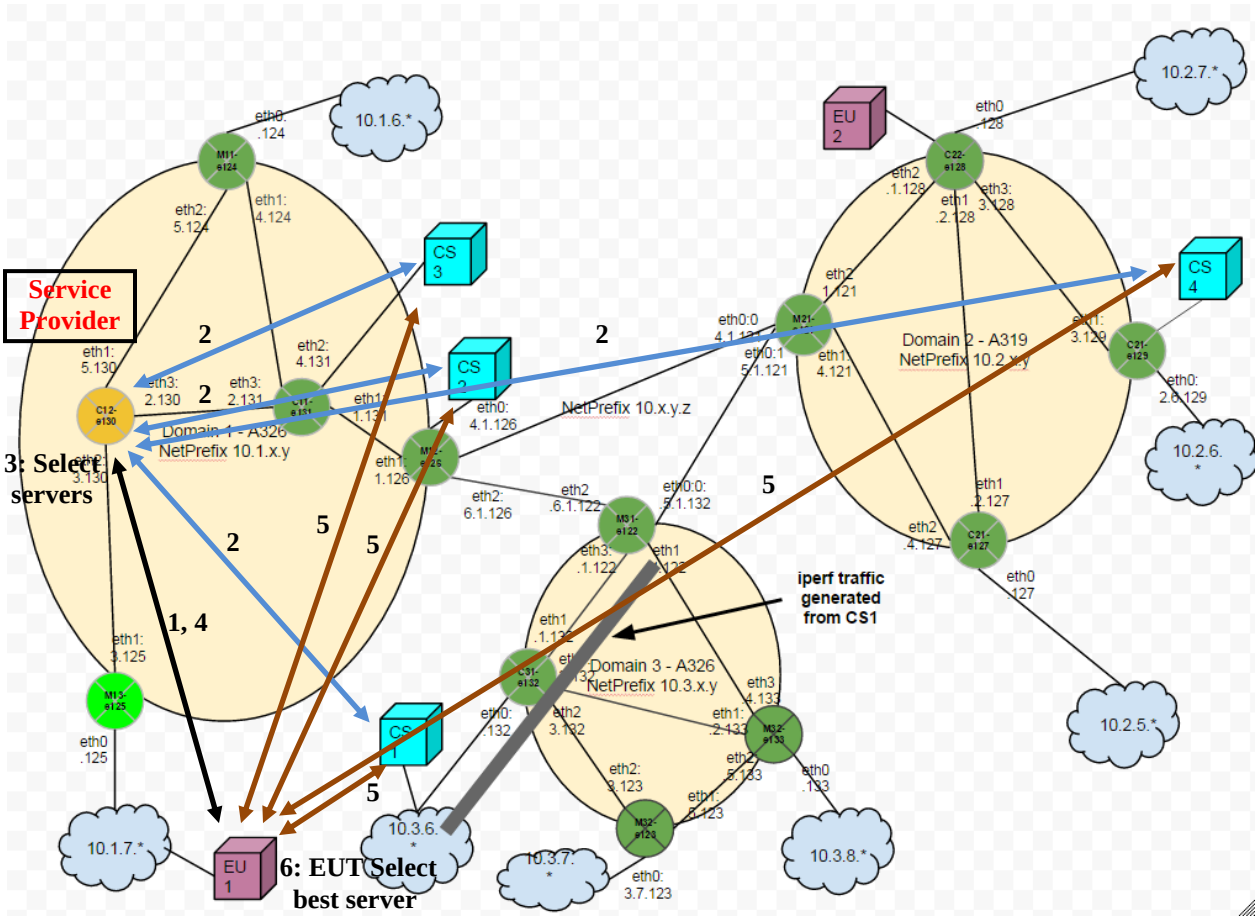

Figure 2. DISEDAN System testbed; interactions between system's modules

The normalization is done such that a small value indicates a busy server, while a big value indicates a free server. The MCDA algorithm uses a **min-max** approach. First, it selects for each server the minimum value among the values of the normalized monitoring parameters for that server. So, it has the worst score for each server. Secondly, it selects the best server as the one that have the greatest score among the scores determined in the first phase.

In this scenario, initially, all the servers' monitoring parameters are similar (because the servers are not loaded), except the network distances which are different. In the first phase the selection, is decided by the network distance. The closest server is chosen as the best one.

In the second phase, the terminal triggers another request for the same service. Prior to this action, the CPU on the server which was selected in the first phase is loaded using the ***stress*** Linux application. The CPU is loaded at around 55%. Consequently, the value of the parameter associated with the CPU load in the MCDA matrix will decrease, forcing the MCDA algorithm to select another best server from the list of servers hosting the requested resource.

D. Validation of the server selection algorithm in different network load conditions

This scenario is similar with the previous one; the difference consists in the parameter used to influence the server selection algorithm. The load on the content server network interface will be used to influence the selection. In this case, the MCDA will be forced to select the servers CS1 and CS4 in the first phase, by decreasing the network distance between EUT and CS1/CS4. The CS1 will be the recommended server due to the lowest network distance. Then, the CS1 sending interface will be loaded with traffic. *Iperf* application will be used to load CS1's interface, by sending traffic from CS1 to the router *e122* (Figure 2).

After *iperf* application is started, a second request for the same resource will be initiated by EUT1. Due to the *iperf* traffic, the monitoring parameter *network interface load* will increase significantly on server CS1, which determines the associate MDCA parameter to decrease accordingly. As a consequence, the MCDA algorithm will select CS4 server, which was the second on the list, as the best serving server to provide the requested resource.

 ISBN: 978-1-61208-455-8

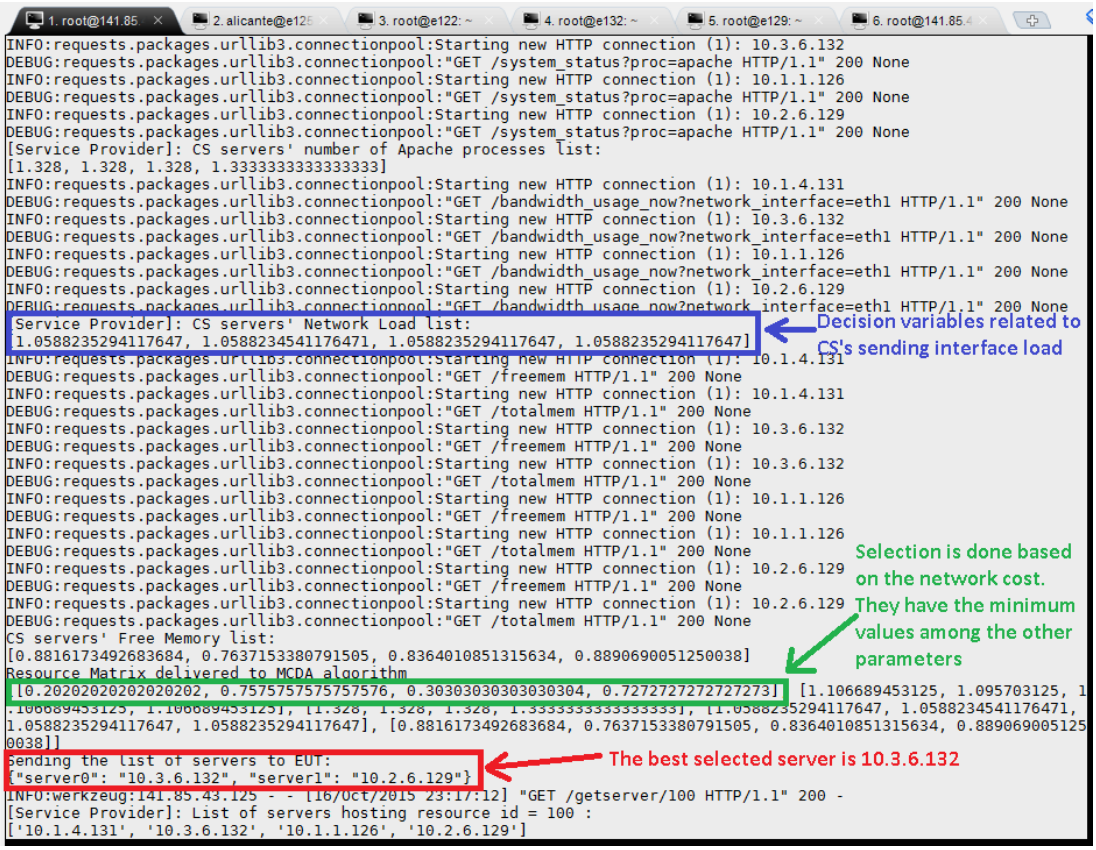

Figure 3. CS selection at SP when the servers are not loaded

In the first phase of this scenario, the content servers are not loaded, which means that decision variables will have similar values. In this case, the selection is done based on the values for the variable representing the network distance between EUT and CSs. The minimum network distance is allocated for CS1 (10.3.6.132), second on the list, followed by CS4 (10.2.6.129), fourth on the list. In Figure 3, the values for the decision variables related to the traffic load on the sending interface for each server hosting the requested resource are shown. Because there is no load on any of the interfaces, the values are similar.

With green there are emphasized the values for the variables related to network cost. They have the minimum values among the values of the parameters in the resource matrix used by MCDA, which means that the selection is done based on these values. As it can be seen, the maximum value among them is for the server 10.3.6.132, which is recommended as the best server, as is emphasized with blue on Figure 3.

In the second phase, the sending interface of the CS1 server is loaded with background traffic, which is generated with ***iperf*** application and is sent to the e122 machine, as can be seen in Figure 2. In this case, the value of the decision variable related to the load on the CS1's sending interface will decrease, determining the MCDA to select as the best server the CS4 machine, with the IP 10.2.6.129. This is illustrated by the SP's messages captured Figure 4. As it is emphasized with blue, the sending interface load decision variable for server CS1 (10.3.6.132) decreased (the second position in the array - its value is around 0.674), determining that this variable will be selected to represent CS1. It also determines the selection of the CS4 server as the best one, due to the ***min-max*** approach in server selection algorithm. So, in the second phase, the selected server is CS4 (10.2.6.129), because it has all the decision variables bigger than 0.674, which is the value of the decision variable selected to represent CS1 - as can be seen in the resource matrix delivered to MCDA in Figure 4. CS4 decision variables are on fourth position in each vector, the servers' order being shown in the last row of messages captured in Figure 3.

V. CONCLUSIONS AND FUTURE WORK

In this paper, a functional validation for the multimedia delivering system proposed in the framework of the DISEDAN project, is presented. A media streaming system, working in OTT style has been considered. To improve the QoS/QoE, the system combines an initial (Phase1) optimized content server selection (based on multi-criteria decision algorithms- MCDA) with in-session (Phase 2) media flow adaptation or content server switching.

The results presented in this paper illustrate the basic functionalities of the DISEDAN system's modules and the operating mode of the MCDA algorithm in the first phase of the content server's selection process, in different content servers and network load conditions.

 ISBN: 978-1-61208-455-8

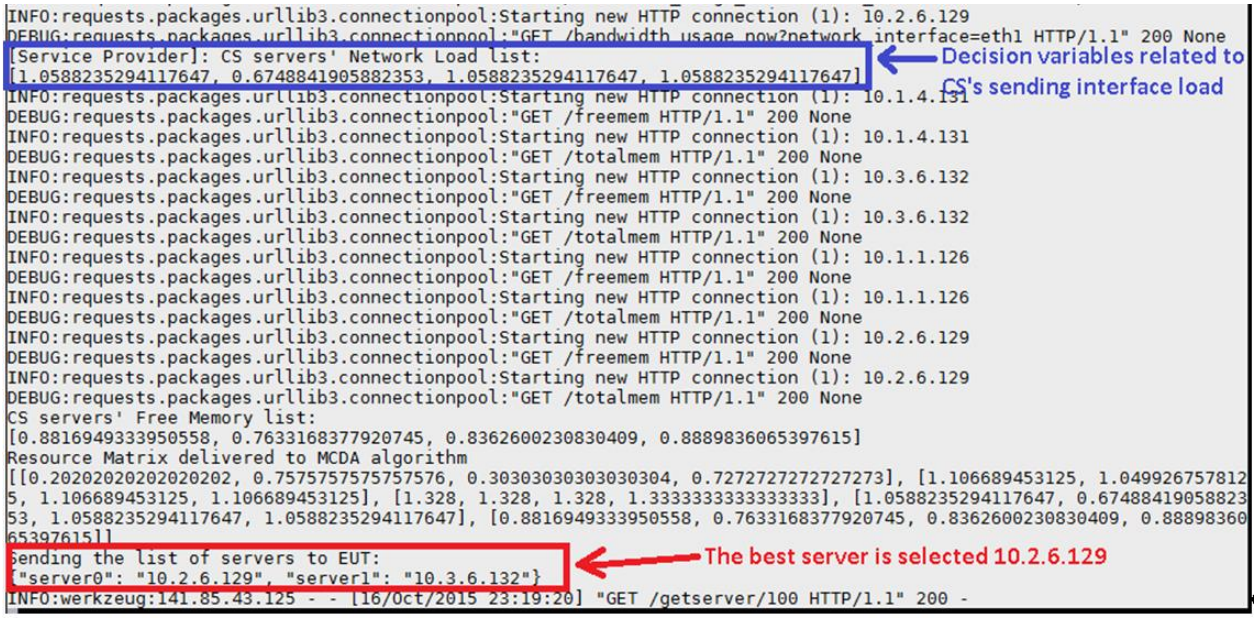

Figure 4. CS selection at SP when the sending interface on the best CS is loaded with traffic.

The experimental results obtained on the real life testbed validated the capability of the system to properly select the "best server", thus efficiently contributing to prepare the media session phase. A more complete set of results regarding the first phase of the server selection were obtained on a simulation model developed for the DISEDAN system and are presented in [15].

More elaborate tests to find some quantitative values for the QoE improvements are under investigation and will be developed in order to take advantage of the real life prototype available. Also, investigation on the second selection phase and the adaptation profess will be performed.

ACKNOWLEDGMENT

This work has been partially supported by the Research Project DISEDAN, No.3-CHIST-ERA C3N, 2014- 2015.

REFERENCES

[1] J. Choi, J. Han, E. Cho, T. Kwon, and Y. Choi, "A Survey on Content-Oriented Networking for Efficient Content Delivery", IEEE Communications Magazine, March 2011, pp. 121-127.

[2] *** http://wp2.tele.pw.edu.pl/disedan/ [retrieved:May, 2015]

[3] T. Dreier, "Netflix sees cost savings in MPEG DASH adoption," 15 December 2011. [Online]. Available: http://www.streamingmedia.com/Articles/ReadArticle.aspx?ArticleID=79409. [retrieved: October, 2014].

[4] I. Sodagar, "The MPEG-DASH Standard for Multimedia Streaming Over the Internet," MultiMedia, IEEE, vol. 18, no. 4, 2011, pp. 62 - 67.

[5] ISO/IEC 23009-1, "Information technology -- Dynamic adaptive streaming over HTTP (DASH) -- Part 1: Media presentation description and segment formats," ISO/IEC, Geneve, second edition, 2014.

[6] ETSI TS 126 247 V11.7.0 (2014-07) (UMTS); LTE; Transparent end-to-end Packet-switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH), (3GPP TS 26.247 version 11.7.0 Release 11, 2014).

[7] A. P. Wierzbicki, "The use of reference objectives in multiobjective optimization". Lecture Notes in Economics and Mathematical Systems, vol. 177. Springer-Verlag, pp. 468–486.

[8] A. Beben, J. M. Batalla, W. Chai, and J. Sliwinski, "Multi-criteria decision algorithms for efficient content delivery in content networks", Annals of Telecommunications - annales des telecommunications, Springer, vol. 68, Issue 3, 2013, pp. 153-165.

[9] E. Borcoci, M. Vochin, M. Constantinescu, J. M. Batalla, and D. Negru, "On Server and Path Selection Algorithms and Policies in a light Content-Aware Networking Architecture", ICSNC 2014 Conference, http://www.elcom.pub.ro/disedan/docs/ICSNC%202014%20Conf.pdf.

[10] E. Borcoci, C. Cernat, and R. Iorga, "Control Plane Design for a Content Streaming System with Dual Adaptation", AICT 2015 Conference, https://www.thinkmind.org/index.php?view=article&articleid=aict_2015_6_40_10154

[11] E. Borcoci, ed., et al., "D2.1 System requirements and comparative analysis of existing solutions for media content server selection and media adaptation", DISEDAN Project, July 2014, http://wp2.tele.pw.edu.pl/disedan.

[12] J. Mongay Batalla, ed. et.al., , "D2.3 Specification of the Dual adaptation mechanism," DISEDAN Project, 2014.

[13] ***, http://www.postgresql.org, [retrieved: March, 2015].

[14] ***, http://nodejs.org, [retrieved: March, 2015].

[15] O. Catrina, E. Borcoci, and P. Krawiec, "Two-Phase Multi-criteria Server Selection for Lightweight Video Distribution Systems," 27th IFIP TC7 Conference 2015 on System Modelling and Optimization Integration of Optimization, Modeling and Data Analysis for Solving Real World Problems.

 ISBN: 978-1-61208-455-8