# Use of Mobile Network Analytics for Application Performance Design

Irene Alepuz, Jorge Cabrejas, Jose F. Monserrat
Universitat Politècnica de València, iTEAM
Valencia, Spain
Email: {iralbe, jorcabpe, jomondel}@iteam.upv.es

Alvaro G. Perez, Gonzalo Pajares, Roberto Gimenez
EUROB CREATIVE
Madrid, Spain
Email: {aperez, gpajares, rgimenez}@eurob.com

***Abstract*—With the 5G technology, data traffic is going to grow by a factor of 1000, while the number of connected devices is likely going to be two orders of magnitude higher. With smartphones being cornerstone in our daily lives, understanding mobile network performance is critical for providing a superior user experience and, consequently, determining the success of an application. This paper presents a solution that uses the radio parameters measured by a mobile terminal to determine the best Application Protocol (APPP) for a service, so as it could adapt to the varying network conditions. From the training of an inference system with actual Mean Opinion Score (MOS) data, it will be possible to discern which radio Key Performance Indicators (KPIs) are best suited to characterize the state of the network and make the best possible decision. Results show how the decision system based on only three radio KPI is able to determine the user application experience with a success of up to 83%. Thanks to the use of this approach, application developers may fill the gap of knowledge between network KPIs and user experience.**

## I. Introduction

The increase of mobile traffic data has led researchers to develop Fifth Generation (5G) of mobile and wireless communication systems. The 5G technology is expected to be a reality by the year 2020 or even sooner. In the meantime, its main characteristics have been described by diverse projects around the world, as the Mobile and wireless communications Enablers for Twenty-twenty (2020) Information Society (METIS) [1] and 5G Novel Radio Multiservice adaptive network Architecture (5G-NORMA) [2] European Commission's projects. In 2020 and beyond, it is expected a 1000x increase of data traffic, including mostly indoor and small cell/hotspot traffic, and a higher number of connected devices that boost the well-known concept of Internet of Things (IoT) [3].

In addition, it is necessary to make a proper design of the Applications (APPs) in order to ensure that they offer the user a good experience, with quick response to their different actions and low consumption of data. Only this way, new applications can gain in reputation and find its place in the daily use of customers. Therefore, understanding mobile network performance is critical for providing this superior user experience.

On the one hand, for users, it is becoming very popular the usage of measurement network tools (like Netalyzr [4] or MobiPerf [5]) to determine problems in their connectivity. However, on the other hand, for software developers, this approach has not yet become a standard procedure when debugging the APPs' performance. There are several well-know tools like Yahoo Mobile, Mixpanel or Countly, which analyse the user behaviour when using an APP. In these tools, metrics like frequency usage, time spent and used functionalities are analysed. These tools specifically focus on the user interaction with the APP and do not provide data for an effective APP tuning in terms of protocols, security, network usage, connectivity strategies, etc. from a pure mobile network approach. Therefore, there is an evident need for such dedicated tools and related knowledge.

With respect to performance metrics, most of the works only take into account pure radio aspects, power consumption or performance bandwidth comparisons among the different available networks. In [6] it is shown the need to optimize the Quality of Experience (QoE) of mobile APPs. In this case, this issue is addressed from a pure radio network quality point of view. In [7], the optimal radio design of the Radio Resource Control (RRC) state machine of smartphones in real scenarios is addressed. This paper concludes that traffic patterns impose significant impact on radio and energy consumption. Then, [8] shows that characterizing and understanding the performance of today's cellular networks is far from trivial and that operators do exhibit significant variance in end-to-end performance in terms of latency and throughput.

Accordingly, this paper, making also use of radio Key Performance Indicators (KPIs), incorporates the user end-to-end experience metric in the form of the Mean Opinion Score (MOS). With this information, the APP will be able to select the most appropriate communication protocol adapted to the radio conditions, for improving user's experience. This paper aims at determining which are the KPIs that govern an APP user experience considering both protocol and radio issues through an Android library, called Napplytics, that could be easily incorporated to any Android APP.

This work is part of the Measuring Mobile Broadband Networks in Europe (MONROE) European Commission's project, whose development platform is used to calibrate and correct the measurements and results provided [9]. Thanks to the use of an actual measurement platform, reliable values of network operation can be obtained, thus resulting in a point of operation of the decision makers close to reality.

The structure of this paper is as follows. In Section II the

Napplytics algorithm is described. Then, results are presented and discussed in Section III. Finally, Section IV draws the main conclusions of the paper.

## II. Napplytics description

Napplytics has been designed to be an Android library with the objective of being a measurement network tool that can be easily incorporated to any Android APP. This Android library is not only a measurement network tool, but also a customer-centric monitoring solution that will add a special value to all the APPs. This library will select the most appropriate Application Protocol (APPP) depending on the network conditions experienced by the user. For example, as it is the case study of this paper, in case the user wants to do a web browsing, this library will configure which would be the optimal parameter configuration to receive the website in the users mobile phone based on its current coverage. It is important to note that this configuration selection will be transparent for both: the APP users and the APP developers. In line with this library, an informative APP is going to be designed. This APP will inform the user which will be the best protocol as a function of the operation realized. In addition, it will inform about the user perception while using different protocols. Note that this APP is out of the scope of this paper. Therefore, this paper, and specially this section, focuses on the algorithm designed to select the best APPP.

Big data and data analytics are currently having an important role in the mobile broadband networks operation and management. In particular, the trend is to store an enormous amount of data related to different KPIs of users and/or their Global Positioning System (GPS) in order to evaluate the network performance. However, these KPIs are not capturing the actual perception of customers, as evidenced by the fact that, recurrently, operators see that network KPIs are very satisfactory but customer care service is facing an increasing number of complaints about poor QoE. Therefore, better models are needed to capture the relationship between QoE and the network KPIs which will inform about the Quality of Service (QoS).

In this sense, this paper proposes a powerful customer-centric monitoring solution that correlates radio network metrics and end-user perception information. Customer experience metrics such as blocked connections, dropped connections, coverage issues and data quality are correlated with technical radio measurements like received pilot power, throughput, duration of the connection, and so on. The result is an advanced customer Satisfaction Index (SI) that not only provides the actual customer perception but also is used to identify the most appropriate communication mode (e.g. Hyper Text Transfer Protocol (HTTP), Real-Time Protocol (RTP), User Datagram Protocol (UDP) or Transmission Control Protocol (TCP)) to adjust the application operation and enhance customer experience. In this case, the paper focuses on three APPPs, which are: HTTP1.1, HTTP2 and HTTP1.1 Transport Layer Security (TLS).

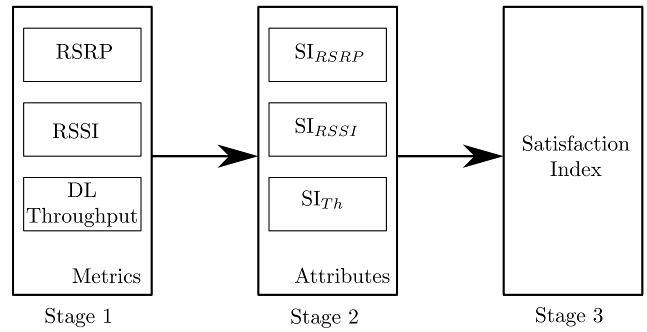


Fig. 1. Satisfaction Index procedure.

The process is divided into three main stages (see Fig. 1): definition of the main network metrics, calculation of the attributes, and finally, calculation of the SI.

### A. Stage 1: Metrics definition

In order to simplify, we give only details for the Long Term Evolution (LTE) technology, although a similar approach can be followed by other technologies, like Universal Mobile Telecommunications System (UMTS) or Global System for Mobile Communication (GSM). Therefore, in this case, the network metrics used are: Reference Signal Received Power (RSRP), Received Signal Strength Indication (RSSI) and Downlink (DL) Throughput (Th). There were more parameters obtained through the MONROE experimental platform, but they were discarded due to dependencies with other selected parameters. As an example, one of the discarded parameters was the Reference Signal Received Quality (RSRQ), which can be derived directly from RSRP and RSSI values.

Once all these metrics are collected, they will be transformed into attributes.

### B. Stage 2: Attributes calculation

From these metrics, specific thresholds are used in order to obtain each attribute, which are the basic components of the final SI. Each attribute, as it can be seen in Table I, has five levels of quality: from 1, the lowest quality, to 5, the highest quality.

The attribute for the RSRP metric in dBm, is calculated as

TABLE I
User experience quality ratings.

| Score | User experienced quality |
|---|---|
| 5 | Excellent |
| 4 | Good |
| 3 | Fair |
| 2 | Poor |
| 1 | Bad |

follows:

$$SI_{RSRP} = \tag{1}$$
$$= \begin{cases} 1, \text{for } RSRP \le -109 \\ f_1^{RSRP}, \text{for } RSRP \in \,]-109,-103] \\ f_2^{RSRP}, \text{for } RSRP \in \,]-103,-97] \\ f_3^{RSRP}, \text{for } RSRP \in \,]-97,-92] \\ f_4^{RSRP}, \text{for } RSRP \in \,]-92,-88] \\ 5, \text{for } RSRP > -88 \end{cases},$$

where $f_1^{RSRP}$, $f_2^{RSRP}$, $f_3^{RSRP}$, and $f_4^{RSRP}$ are linear functions inside its own interval. For example,

$$f_1 = 1 + \frac{RSRP + 109}{6}. \tag{2}$$

A similar calculation is carried out for the rest of the metrics: RSSI ($SI_{RSSI}$) and DL throughput ($SI_{Th}$):

$$SI_{RSSI} = \tag{3}$$
$$= \begin{cases} 1, \text{for } RSSI \le -81 \\ f_1^{RSSI}, \text{for } RSSI \in \,]-81,-76] \\ f_2^{RSSI}, \text{for } RSSI \in \,]-76,-71] \\ f_3^{RSSI}, \text{for } RSSI \in \,]-71,-66] \\ f_4^{RSSI}, \text{for } RSSI \in \,]-66,-60] \\ 5, \text{for } RSSI > -60 \end{cases},$$

$$SI_{Th} = \tag{4}$$
$$= \begin{cases} 1, \text{for } Th \le 0 \\ f_1^{Th}, \text{for } Th \in \,]0,500] \\ f_2^{Th}, \text{for } Th \in \,]500,800] \\ f_3^{Th}, \text{for } Th \in \,]800,900] \\ f_4^{Th}, \text{for } Th \in \,]900,1300] \\ 5, \text{for } Th > 1300 \end{cases},$$

as in the $SI_{RSRP}$ case, the $f_1^{RSSI}$, $f_2^{RSSI}$, $f_3^{RSSI}$, $f_4^{RSSI}$, $f_1^{Th}$, $f_2^{Th}$, $f_3^{Th}$ and $f_4^{Th}$ are linear functions inside its own interval. In order to obtain these intervals, about 700,000 samples have been taken into account. These samples were distributed into five different levels as described in Table I. The span of each level was chosen so as to guarantee a similar amount of samples per level.

### C. Stage 3: SI Calculation

The final SI result, which corresponds to a measure of the MOS of the customer concerning the service provided by the operator, is calculated as a function of the attributes defined for the specific technology. For this function, we used a weighted sum of attributes as follows:

$$SI = w_1 SI_{RSRP} + w_2 SI_{RSSI} + w_3 SI_{Th}. \tag{5}$$

In the training phase, the weights ($w_1$, $w_2$ and $w_3$) are calculated. Note that the objective of the SI is to describe the relation between the network metrics KPIs and the QoE perceived by the user of a service. In this sense, the optimum weights for the components of the SI attributes that fulfill that relation can be directly calculated through customers surveys. However, this method implies an enormous effort that usually it is not possible to handle. Thus, a common approach to simplify the process of measuring the QoE is by means of utility functions, which capture the trends of customer's surveys in mathematical form. These utility functions have been widely studied and used in the literature [10], [11], [12], [13], [14].

For the sake of simplicity, this paper focuses on only web browsing services, so the utility function that fits their MOS is presented in [10] as:

$$U = 5 - \frac{578}{1 + \left(11.77 + \frac{22.61}{d}\right)^2}, \tag{6}$$

where $d$ stands for the service response time measured in seconds.

Thus, the weights ($w_1$, $w_2$ and $w_3$) are calculated by minimizing the Mean Square Error (MSE) of the difference between the SI values and the values of the utility function ($U$). Letting $U$ and $SI$ be now random variables, this will result in the following:

$$\min_{w_1,w_2,w_3} \mathbb{E}[(U - SI)^2]. \tag{7}$$

This problem was solved using the Generalized Reduced Gradient (GRG) algorithm [15], which is a generalization of the reduced gradient method by allowing nonlinear constraints and arbitrary bounds on the variables.

Since each service has different performance targets, each service has different utility functions. In this work, different applications have been tested for the web browsing service so as to select that application with the highest SI. The objective is to experiment, while using different APPP, the best performance when executing a service (e.g., Web Browsing, Youtube, Bittorrent, etc). The calculation of the different SIs that represents the relation between the network QoS and the user QoE are described in Fig. 2. The calculation of the SIs is based on training, in which we know some application layer metrics, like service response time. For each service $i$-th, the inference system executes the following steps:

1) To list the $M$ APPPs that can support this service (APPP 1, APPP 2, $\cdots$, APPP $j$, $\cdots$, APPP $M$), being $M$ specific to each service.
2) To obtain the set of utility values for each APPP using all experimental measures related to this service and APPP using Eq. 6, yielding the set of vectors: $\mathbf{u}_i^1, \mathbf{u}_i^2, \cdots, \mathbf{u}_i^j, \cdots, \mathbf{u}_i^M$.
3) To collect all attributes used in the calculation of the $SI$, in this case, $\mathbf{SI}_{Metric1}$, $\cdots$, $\mathbf{SI}_{MetricL}$.
4) Finally, as it has been explained, the weights ($w_1$, $w_2$ and $w_3$) are calculated as in Eq. 7, but with the particularization of being a finite set of samples, minimizing the MSE of the set of $SI$ values of all experiments with respect to their corresponding utility values.

In the execution phase, only the radio information is available, so that the user experience could not be known a priori.

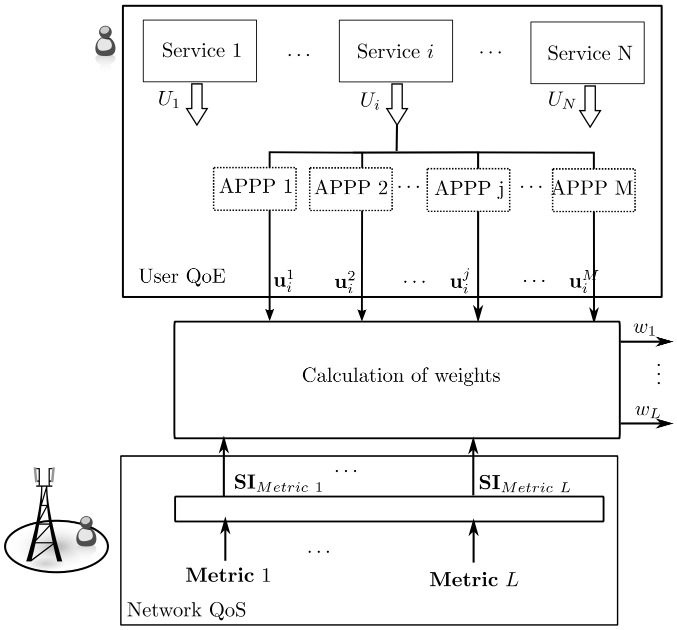


Fig. 2. Calculation of the different SIs for each service.

It is here where the inference system, already trained, can estimate the MOS from the radio attributes, that is, $SI_{RSRP}$, $SI_{RSSI}$ and $SI_{Th}$ that can be instantaneously known, simply by using Eq. 5, and the specific weights derived in the training phase for each APPP.

This process will be executed for each one of the studied APPP. Subsequently, a $SI$ per protocol is obtained. Finally, the selected protocol will be the one that fits:

$$\max[SI_{APPP1}, SI_{APPP2}, \cdots, SI_{APPPM}]. \tag{8}$$

An example of user plane protocol stack using Napplytics method is depicted in Fig. 3. In this figure, it is observed the different protocol entities of the Radio Access Network (RAN) user plane, which are [16]: Physical (PHY), Medium Access Control (MAC), Radio Link Control (RLC), Packet Data Convergence Protocol (PDCP), Internet Protocol (IP), TCP, plus the multiple-option application layer protocols, which will be selected as a function of the varying network conditions.

## III. Results and discussions

In this section, the results obtained with more than 2500 experiments are presented and discussed. In this study case, the web browsing service was analysed using different APPPs. On this occasion, the studied protocols were HTTP1.1, HTTP2 and HTTP1.1 TLS. This service was studied executing a combination of the *headlessbrowsing* and *http_download* experiments provided by MONROE consortium [9]. During these experiments, web pages like Google, Instragram, Facebook or Wikipedia, among others, were downloaded. The experimental nodes were placed in different countries, like Norway, Sweden, Italy or Spain. Also, these nodes had special characteristics. For example, a node could be static or mobile. In these sense, these variety of situations will provide us with a rich and huge amount of data.

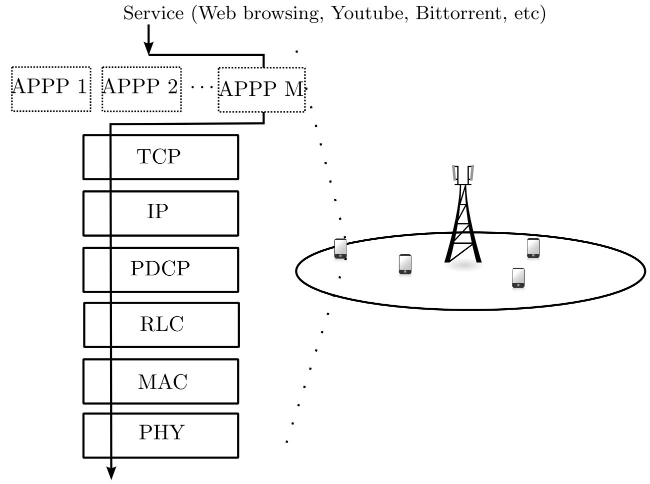


Fig. 3. Napplytics method.

First of all, the results obtained with the HTTP1.1 protocol are analysed. In this case, we have 887 experiments, that is, 887 web downloads. For this case, as it can be seen in the following figures, there are similarities between the distribution of the different results of the SI (Fig. 4) and the ones obtained with the utility function (Fig. 5). As matter of fact, the number of experiments with a value of 3 in the SI are the most common (they are up to 500 experiments). This correlates with the number of experiments with a value of 3 obtained by the utility function ($U$), which are also the most common (they are up to 600 experiments). By contrast, values 1 and 5 have the least number of experiments in both distributions, 26 experiments at the SI distribution and 24 experiments for the utility function distribution.

Fig. 6 shows the difference between $SI$ function and the results obtained by the utility function ($U$). In this figure, the detractor experiments are defined as those that experi-

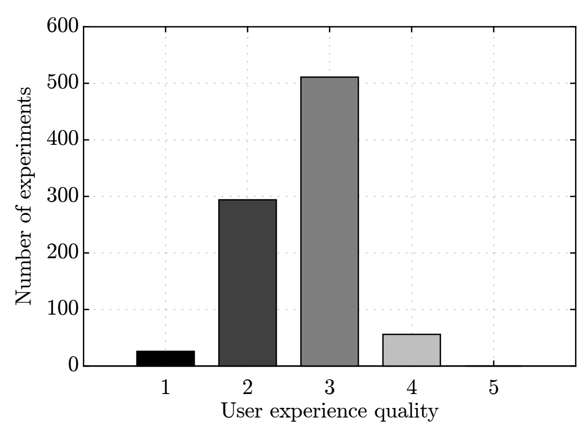


Fig. 4. Distribution of the user experience quality by the $SI$ for the HTTP1.1 protocol.

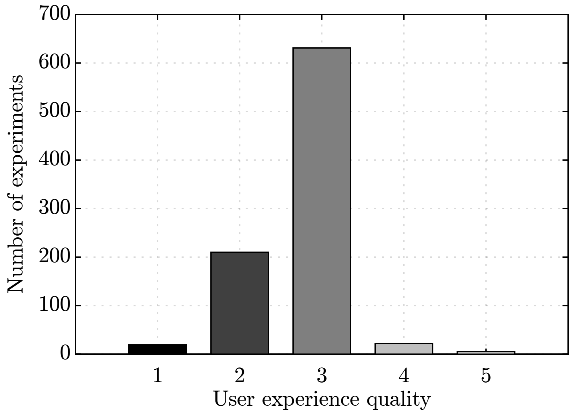


Fig. 5. Distribution of the user experience quality by the $U$ for the HTTP1.1 protocol.

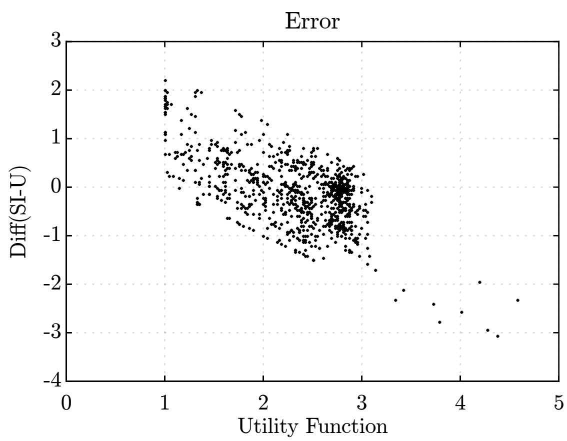


Fig. 6. Error obtained with the protocol HTTP1.1.

enced good network KPIs but still had a low utility value. Mathematically interpreted, we say an experiment is a detractor if the difference between the SI level and the utility function ($U$) level is greater than or equal to 1.5 points. In contrast, a promoter experiment is an experiment that, even experiencing bad network quality KPIs, still obtains good utility values ($U$). Likewise, the promoters experiments are mathematically defined by a difference less than or equal to $-1.5$ points. In the studied case, for HTTP1.1 protocol, there are $3.04\%$ of detractor experiments and $1.47\%$ of promoter experiments. Then, the correlation between SI and utility function is made by considering that they correlate if the difference between them is less than or equal to 1 point. In this case, the correlation is $86.58\%$. If we do not take into account detractor and promoter experiments, the correlation increases up to $90.67\%$. Moreover, the MSE for the HTTP1.1 protocol is $0.5$.

Similarly, we studied the rest of protocols (HTTP2 and HTTP1.1 TLS). In Table II the contribution of each parameter to the $SI$ for each one of the protocols is presented. It can be

TABLE II
WEIGHT PERCENTAGE FOR EACH PARAMETER.

| Parameter | HTTP1.1 | HTTP2 | HTTP1.1 TLS |
|---|---|---|---|
| RSRP | 30.77% | 31.12% | 32.57% |
| RSSI | 28.05% | 28.68% | 27.80% |
| DL Throughput | 41.19% | 39.21% | 39.63% |

TABLE III
PERCENTAGE OF DETRACTORS AND PROMOTERS.

| Detractors | Promoters | Protocol |
|---|---|---|
| 3.04% | 1.47% | HTTP1.1 |
| 1.32% | 1.59% | HTTP2 |
| 1.8% | 1.94% | HTTP1.1 TLS |

seen that, in all protocols, the most dominant parameter is the DL throughput. Then, in Table III the percentage of detractor and promoter experiments are presented for each of the studied protocols. As it can be seen, in all protocols the percentage of detractors and promoters experiments is around $1.5\%$, except for the HTTP1.1 protocol, where detractor experiment raise up to $3\%$. Next, in Table IV the correlation between the SI and the results obtained with the utility function are presented for both cases, with and without taking into consideration the detractor and promoter experiments. As the Table IV shows, the correlation in all cases is above $80\%$, and when detractors and promoters are not considered, the correlation is around $90\%$. Finally, the MSE obtained for each one of the protocols is presented in Table V. Here, it is observed that in all cases the MSE is below $0.6$, being HTTP1.1 protocol the one exhibiting the minimum MSE.

## IV. CONCLUSIONS

This paper has described a new approach to correlate network KPIs with user experience while using an APP on mobile phone. As it is evidenced in the results section, the proposed algorithm exhibits high correlation with the actual MOS in the experiments we analysed.

In future work, more services and protocols, for example, Voice over IP (VoIP) or video streaming, and, RTP or UDP respectively, will be studied. This implies the execution of more experiments in the MONROE experimental framework, but we could take advantage of the experiments that other

TABLE IV
CORRELATION IN THE DIFFERENT PROTOCOLS.

| Correlation | Correlation without D&P | Protocol |
|---|---|---|
| 86.58% | 90.67% | HTTP1.1 |
| 86.9% | 89.51% | HTTP2 |
| 83.72% | 86.98% | HTTP1.1 TLS |

TABLE V
MEAN SQUARE ERROR IN THE DIFFERENT PROTOCOLS.

| Mean square error | Protocol |
|---|---|
| 0.5 | HTTP1.1 |
| 0.53 | HTTP2 |
| 0.59 | HTTP1.1 TLS |

partners will do in the different investigations running in parallel.

Finally, this new approach will help software APP developers to understand the user's experience during the APP execution time and change the application protocol accordingly. Also, in this competitive world of mobile APPs, where for example, just in Google Play there are more than 2 millions of APPs, differentiation from competitors is highly important to guarantee the success of an APP. Therefore, this new approach will improve the APP performance, what will lead to enhance the APPs user's satisfaction as well as the loyalty and engagement with the APP.

## Acknowledgement

This work is funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 644399 (MONROE) through the open call project NAPPLYTICS. The views expressed are solely those of the authors.

## References

[1] J. Monserrat et al., *Rethinking the Mobile and Wireless Network Architecture: the METIS Research into 5G,* Proceedings of European Conference on Networks and Communications (EuCNC), pp. 15 (2014).

[2] H. Droste et al., *An adaptive 5G Multiservice and Multitenant Radio Access Network Architecture,* Transactions on Emerging Telecommunications Technologies, vol. 27, no. 9, pp. 12621270 (2016).

[3] N. Al-Falahy and O. Y. Alani, *Technologies for 5G Networks: Challenges and Opportunities,* IEEE IT Professional Magazine, pp. 12-20 (2017).

[4] C. Kreibich, N. Weaver, B. Nechaev and V. Paxson, *Netalyzr: Illuminating the Edge Network,* Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement, pp. 246-259 (2010).

[5] J. Huang et al., *Mobiperf: Mobile network measurement system,* Technical Report. University of Michigan and Microsoft Research (2011).

[6] F. Kaup, F. Jomrich and D. Hausheer, *Demonstration of Network Coverage - A Mobile Network Performance Measurement App,* Proceedings of the IEEE International conference on Networked Systems (NetSys) (2015).

[7] F. Qian, Z. Wang, A. Gerber, Z.M. Mao, S. Sen and O. Spatscheck, *Characterizing Radio Resource Allocation for 3G Networks,* Proceedings of the 10th ACM SIGCOMM conference on Internet Measurement, pp. 137-150 (2010).

[8] A. Nikravesh, D.R. Choffnes, E. Katz-Bassett, Z.M. Mao, and M. Welsh, *Mobile Network Performance from User Devices: A Longitudinal, Multidimensional Analysis,* Proceedings of the International Conference on Passive and Active Network Measurement, pp. 12-22 (2014).

[9] V. Mancuso et al., *System Design and Prototype Implementation,* in *Measuring Mobile Broadband Networks in Europe (MONROE)* D1.2 (2016).

[10] P. Ameigeiras and J. J. Ramos-Munoz, J. Navarro-Ortiz, and P. Monensen *QoE Oriented Cross-layer Design of a Resource Allocation Algorithm in Beyond 3G Systems,* Computer Communications, vol. 33, no. 5, pp. 571-582, Elsevier (2010).

[11] R.K.P. Mok, E.W.W. Chan and R.K.C. Chang *Measuring the Quality of Experience of HTTP Video Streaming,* Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 485-492 (2011).

[12] J. Navarro-Ortiz, J. Lopez-Soler and G. Stea *Quality of Experience based Resource Sharing in IEEE 802.11e HCCA,* Proceedings of the European Wireless Conference (EW), pp. 454-461 (2010).

[13] P. Oliver-Balsalobre, and M. Toril, S. Luna-Ramírez and J.M. Avilés *Self-tuning of Scheduling Parameters for Balancing the Quality of Experience among Services in LTE,* EURASIP Journal on Wireless Communications and Networking, pp. 1-7 (2016).

[14] G. Gómez, J. Lorca, R. García and Q. Pérez *Towards a QoE-driven Resource Control in LTE and LTE-A Networks,* Journal of Computer Networks and Communications (2013).

[15] J. Abadie, *The GRG Method for Nonlinear Programming,* Design and Implementation of Optimization Software, Springer Netherlands, vol. 28, pp. 335-363 (1978).

[16] E. Dahlman, S. Parkvall and J. Sköld, *4G: LTE/LTE-Advanced for Mobile Broadband,* in *Elsevier* (2014).