MULTIMEDIA COMMUNICATIONS TECHNICAL COMMITTEE
http://www.comsoc.org/~mmc

MMTC Communications - Frontiers

Vol. 14, No. 5, September 2019

CONTENTS

SPECIAL ISSUE ON New Frontiers in Edge Computing Research

Guest Editor: Luca Foschini
University of Bologna, Italy
luca.foschini@unibo.it

This special issue of Frontiers sheds light on new perspectives and latest achievements in edge computing research. In fact, fueled by recent advances in wireless communication technologies, resource virtualization technologies, and cloud management, we are witnessing a variety of new application scenarios where edge computing is successfully employed, ranging from Internet of Things (IoT) and sensing, to mobile entertainment, from smart health to Industry 4.0.

The first paper of the issue focuses on energy-neutral management of IoT devices at the edge. Chessa *et al.* propose a management solution able to carefully schedule all tasks to be executed at the IoT node; the authors developed and deployed the proposed solution in a real experimental testbed that includes needed hardware (i.e., energy-harvesting device) and software modules (i.e., energy-neutral scheduler with weather forecast awareness). A large set of experimental results demonstrate the feasibility of the proposed prototype.

In the second paper, Laso *et al.* present a support environment to emulate the execution of Android-based apps and the automatic generation of skeleton proxies to facilitate the offload of parts of the computation to edge nodes. That is particularly beneficial in current edge computing scenarios that are highly heterogeneous, including devices sometimes with stringent resource constraints, various operating systems and execution platforms, and different Application Programming Interfaces (APIs). Reported experimental results confirm the feasibility of the proposed emulator and its usability in real deployment scenarios.

The third paper, by Vitello *et al.*, focuses on the effective deployment of Multi-access Edge Computing (MEC) Edge Data Centers (EDCs) in urban environments. The authors propose a methodology based on the use of mobility of citizens and their spatial patterns; in particular, the proposed solution proposes to combine and use in an innovative way social network tools and datasets, such as the mobility traces from Google Popular Times, by integrating them with specific simulation environments, such as the CrowdSenSim simulator proposed by the same authors. The authors also report a large set of results to show the advantages obtainable using the proposed solution.

Finally, I want to thank all the people involved in the effort of producing this issue, and especially the submitting authors. My gratitude and recognition go to all these contributors, it was a pleasant experience for me, and I hope you will find this special issue informative and useful.

Luca Foschini [SM] (luca.foschini@unibo.it) is an Associate Professor of computer engineering at the University of Bologna. He graduated from the University of Bologna, Italy, where he received a Ph.D. degree in computer engineering in 2007. His interests include distributed systems and solutions for system and service management, management of cloud computing, context data distribution platforms for smart city scenarios, IoT solutions, Industry 4.0, and Factory Digitalization. He has published over 150 conference and journal papers in these areas, receiving best paper award recognitions from various IEEE conferences and highly-cited paper mentions in IEEE journals. His research has been sponsored by local regional funds and industrial companies. He has served as reviewer for several IEEE, Elsevier, and Wiley journal venues, and is also a member of the Editorial Boards of IGI IJHCR and IJARAS, and Hindawi IJDSN and WCMC. He has been actively participating in international conferences as TPC chair. He is a senior member of IEEE and a member of ACM.

Towards an Internet of Energy-Neutral Things

Stefano Chessa[1]*, Soledad Escolar*[2]*, Antonio Caruso*[3]*, Xavier del Toro*[2] *and Juan Carlos López*[2]*,*
[1]*Department of Computer Science, University of Pisa, Pisa, Italy.*
[2]*School of Computer Science, University of Castilla-La Mancha, Ciudad Real, Spain.*
[3]*Department of Mathematics and Physics 'Ennio De Giorgi', University of Salento,, Lecce, Italy*
stefano.chessa@unipi.it; Soledad.Escolar@uclm.es; antonio.caruso@unisalento.it;
xavier.deltoro@uclm.es; JuanCarlos.Lopez@uclm.es;

1. Introduction

The pervasive use of Internet of Things (IoT) technologies, which is changing many aspects of human activities, is traditionally associated to the use of low-power, low cost devices typically embedding simple sensors. However, an important fraction of IoT applications also include more powerful devices (although not as powerful as desktop devices), embedding more powerful sensors like cameras, microphones etc. Examples are surveillance cameras used to monitor infrastructures in smart cities or industries etc. If, on one hand, those devices should support complex applications and require adequate computational power and memory, and a corresponding energy budget, on the other hand, their deployment outdoor (as it is the case of network of surveillance cameras in a smart city) brings the same issues encountered in the outdoor deployment of any IoT solution concerning the management of the energy of such devices and the reduction of the costs concerning their deployment and maintenance. It is common, for this reason, to adopt solutions based on energy harvesting (usually by means of photovoltaic panels) to provide adequate power to such devices. However, this proved by itself insufficient, if not combined with strategies for the management of the power that guarantee properties of energy neutrality to the devices.

Namely, an energy-harvesting device is energy neutral if it modulates its load (i.e. its power consumption) in order to match the production of energy over a given time frame. For example, this can be achieved with a photovoltaic energy harvesting device by guaranteeing that, over each single day, the overall energy consumption is not above the overall energy production. In an IoT device operating with multimedia sensors, this can be achieved by defining different alternative tasks (for example, tasks sampling at high or at low rate, with high or low resolution, etc.), and by scheduling the tasks according to the energy budget and to the expected energy production over the day. Clearly, scheduling a task that samples a camera at high rate with high resolution would be more energy expensive, but it would also provide a better quality of service. The aim of the scheduling is thus to find an optimal tradeoff between energy consumption and quality of service while guaranteeing the energy neutrality of the device.

This work summarizes the works in this field and discusses how the availability of fresh information about weather forecast through the Internet can be exploited to implement the energy-neutral scheduling strategies. In particular, we built an experimental testbed composed of two independent devices. One device acts as an IoT node that can be programmed with different tasks to experiment with different behaviors (and thus different energy consumptions of the device), and one device acts as monitor of all the parameters concerning the energy production of the photovoltaic panel and of the battery charge, without interfering with the normal operations of the first device. With this testbed we have initiated the collection of a dataset comprising the above described parameters concerning energy consumption and production, and local weather information that, combined with weather forecasts from web broadcasters can be used to experiment different energy-neutral scheduling strategies.

2. Energy-Neutral Schedulers

Energy neutrality is the ability for keeping indefinitely the device operation by means of some energy harvester and some optimization strategy that aligns the workload allocation with the energy availability. In a given period of time the consumed energy in the period is always lower or equal than the energy produced in the same period. The first approach is original from Kansal and Srivastava who proposed EEHF (Environmental Energy Harvesting Framework) [1], later refined in [2].

Following the way paved by Kansal and Srivastava, we have introduced a slightly different model, in which the IoT device has available several alternative tasks, each characterized by a different Quality of Service (QoS), duty cycle and energy consumption. Hence the problem of energy neutrality becomes a problem of scheduling of the tasks over a sequence of time slots over a given period (typically of 24 hours for photovoltaic energy harvesting devices), so to maximize the QoS while keeping the device energy neutral. For example, considering a IoT device that implements a

surveillance camera, a task could be the sampling at high rate/high resolution of the camera and the real-time

TABLE I
COMPARISON OF SCHEDULERS AIMED AT ACHIEVING ENERGY-NEUTRALITY (COLORED ROWS CORRESPOND TO OUR WORKS).
LEGEND: N: NUMBER OF SENSORS. **S:** NUMBER OF SINKS. [1]: METEOROLOGICAL ADAPTATION. [2]: UNINTERRUPTED EXECUTION.

Strategy	Workload Model	Energy Source	Simulation	Experimentation			Optimal	Adaptive[1]
				Platform	Scale	Autonomy[2]		
EEHF [1]	Duty Cycle	Solar	N = 100	No			No	Yes
Kansal [3]	Duty Cycle	Solar	N = 100	Heliomote	N = 1	67 days	Yes	Yes
LSA [4]	Sampling & Transmission rates, No. of executed operations	Solar	N = 1	BTnode	N = 1	0 days (just overhead)	Yes	No
Escolar et al. [5]	Sampling frequency	Solar	N = 1	No			No	No
Escolar et al. [6]	Quality	Solar	N = 1	No			No	Yes
Escolar et al. [7]	Quality	Solar	N = 1, S = 1	No			No	No
Escolar et al. [8]	Quality	Solar	N = 1, S = 1	No			No	Yes
Escolar et al. [9]	Quality	Solar	N = 9, S = 1	No			No	Yes
Caruso et al. [10]	Utility	Solar	N = 1	Raspberry PI, Arduino, Tmote	N = 1	0 days (just overhead)	Yes	Yes
Escolar et al. [11]	Utility	Solar & Wind Speed	N = 1	No			Yes	No

transmission of the entire video stream to the cloud (this could be a high-quality, high-cost task), while another alternative task could be one that samples a high rate but with average resolution, that performs locally some preprocessing to extract relevant features, and that transmits only the features.

On the base of this model, we have considered progressively richer scenarios, from a simple one with one energy harvesting device [5], [6], to scenarios involving more devices connected to a gateway or sink, that also has energy harvesting capability [7], [9]. Figure 1 shows a simulation of how the IoT devices and the sink achieve the energy neutrality condition. These initial works adopted a greedy algorithm for the scheduling, so to minimize the execution time and to be feasible even on low-end IoT devices. In a recent work we considered again the simpler scenario, but we introduced a scheduling algorithm based on dynamic programming that finds the optimum scheduling, and we proved that such an approach is even feasible in low-end devices by considering the limited resolution of these devices in sampling the battery charge and the power production of the photovoltaic panel. Our last work [11] considers a richer scenario where a device adopts several, independent energy harvesting subsystems (like a photovoltaic panel and a wind turbine), and it considers the error between the real energy production and the forecast to weaken the constraint of energy neutrality, by allowing slight deviations to occur. The deviations can then be recovered in the next period.

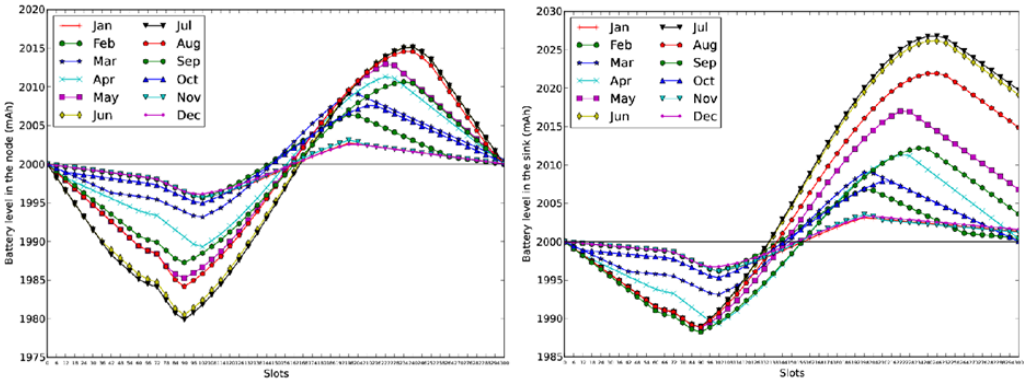


Figura 1 Battery levels in the sensor (on the left) and in the sink (on the right) for all months of the year [7]

3. Testbed and Datasets

Most of the works pursuing energy neutrality are evaluated by means of simulations that model on one hand, the energy states of the hardware components of the device for estimating its consumption and, on the other hand, the solar energy production that will permit increase (or decrease) its energy availability. The solar energy production is estimated based generally on the ideal daily production, which could be reduced by the presence of clouds, shadows, and other meteorological events that are not usually included in the simulations, which, therefore, lack of the realism that we would find in a real deployment. This fact motivates the need of using real testbeds for experimenting energy-aware solutions and, in parallel, for collecting data of energy production that can serve as basis for algorithms with

different purposes, as the search of the optimal energy-neutral schedulings that maximize the applications utility or the development of precise energy forecasting models. This last purpose is covered in several works that describe real energy solar-harvesting testbeds [2], [12], [13], which are based on the proprietary platform VISE, and the commercial platforms TelosB and Waspmote, respectively. The three works have published the collected datasets as a result of their deployments, further used for feeding the forecast of solar energy. However, as far as the authors know, there exist no previous work that proposes an energy-harvester device specifically designed to experiment different energy-neutral schedulers for IoT devices and to assess models and algorithmic solutions for the management of such devices. So, with this purpose in mind, we have developed an energy-harvesting device based on a photovoltaic solar panel, specifically designed for evaluating different energy harvesting algorithms and Internet-of-Things applications. To this end, the device is composed of two independent, both functionally and energetically, microsystems: an IoT node and a Data Logger node. A detailed description of this device can be found in [14] and [15]. We have used this testbed to conduct a data collection campaign that generated a dataset from the environmental and energy-related data sampled with a period of 1 minute.

The testbed (in Figure 2) was installed in Ciudad Real (Spain) and operated without interruptions during the period between July 31st to October 4th, 2018. The dataset comprises 93438 data records, is public and can be downloaded at `https://github.com/arco-group/energy-harvesting-dataset.git`. In this first data collection campaign, the testbed stored the readings into a memory card.

Figure 3 shows the data collected between August 1th to 3th of October. In a second data collection campaign, our testbed was installed on a vineyard and used a LoRa communication module to transmit the data towards a gateway [16]. In turn, the gateway forwarded the received data through a WiFi connection towards a network server where the data were permanently stored. Figure 4 shows the dashboard designed for the real-time data visualization.

4. System Architecture

Figura 2 Deployment of our energy-harvested IoT prototype on top of a building of University of Castilla-La Mancha, Ciudad Real (Spain).

IoT devices tend to include multiple choices for Internet connection, through for instance WiFi, LoRa, Bluetooth, and SigFox network technologies, as one of the main foundations of IoT is to make people and things to be connected anytime, anyplace, with anything and anyone.

This characteristic is favouring more distributed scenarios where the IoT devices are close to interact with services and resources allocated on remote sites for carrying out their purposes in a better way. In such scenarios, the devices may access to the external network both for obtaining more up-to-date data to perform their computations and for locating these computations on remote nodes that can execute them in a more efficient way, for instance, by minimizing performance metrics as the overhead, the latency, and the energy. The combination of IoT with cloud and fog technologies [17], [18] enable the IoT devices to delegate their workloads and storage needs on nearer edge nodes

resulting in lower response times and overhead. The problem that we are addressing fits especially well here, since an IoT device could on one hand, to emplace the energy-neutral scheduler on an adequate edge node, and on the other hand, to download meteorological forecasts (e.g. temperature, wind speed, rain probability) from different weather online web services that can be used to feed the energy production model.

Figure 4 shows a system architecture that supports the described scenario. We consider a LoRa-based network composed by several of our solar energy-harvesting device acting as end-nodes, each one executing an IoT outdoor

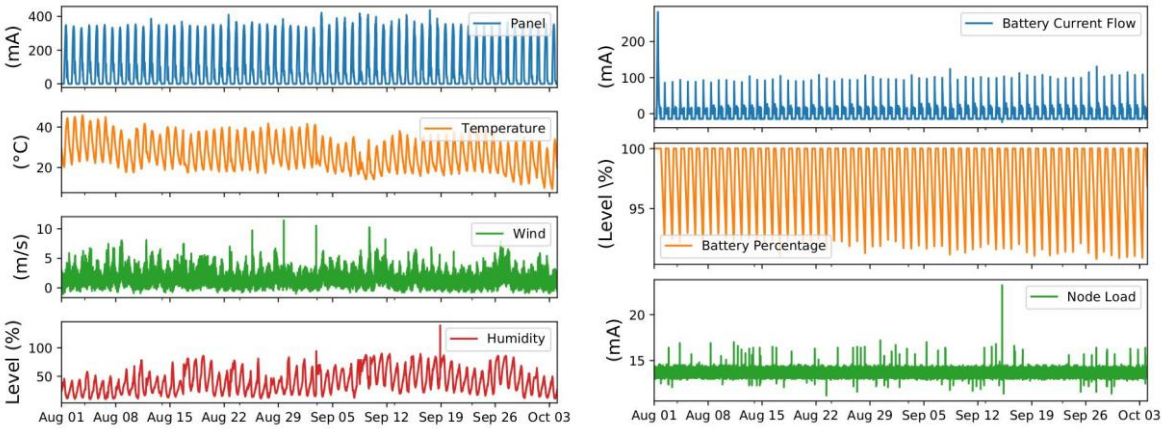


Figura 3 Energy production from the panel, current drawn by the battery, IoT node current, battery voltage, temperature, wind speed and humidity collected during three days of the reference period.

monitoring application as described in [16]. As already mentioned, the device integrates sensors for measuring both meteorological conditions (e.g. temperature, humidity) and energy-related data (e.g. solar production, node consumption, and battery level) and makes usage of its LoRa radio to transmit the samples towards a gateway with capabilities of data logging and wireless connection to Internet. In order to avoid high latencies, to reduce the energy consumption of the IoT device and because of the IoT device could not be able to access directly to the weather online services, the gateway will be in charge of downloading the forecasts from one ore several weather services and storing them into a database. In turn, at the end of each day the IoT device will access the gateway for obtaining the forecasts for the next day previously stored, that are necessary to perform the estimations of the energy production. Alternatively, the IoT device may also decide to centralize or distribute the computations of the schedule between one or several edge/fog nodes based on performance criteria. The correct placement of these computations, in the node itself, in the edge devices or in the cloud, represent an important design decision that we would like to explore in the future.

5. Conclusions and Future Works

The testbed for the collection of dataset concerning energy production and consumption in IoT nodes and the dataset already collected are the first steps of a longer-term research, that aims at identifying proper tradeoffs between QoS

Figura 4 On the left, a dashboard to visualize the data collected by our device. On the right, a three-layered cloud architecture for IoT energy-neutral devices.

of the IoT activities and power consumption within the constraint of energy neutrality of the devices. These tradeoffs depend also on considerations related to the cost and to the computational power of the device. Costs are relevant because they determine physical parameters of the device concerning the battery capacity and size and the photovoltaic panel size and efficiency. The computational power, on the other hand, determine the maximum feasible complexity of the scheduling algorithms that can be employed. Considering IoT devices involved in multimedia data production and processing, which are usually high-end IoT devices, with higher computational and communication performance and memory capacity, this tradeoff may allow, at least in principle, the use of more complex and scheduling algorithms supporting a larger variety of alternative tasks exposing different QoS, and a control of the battery charge and power production with a finer resolution. This research is, however, still in its early phase, as most research efforts so far focused on low power IoT devices. Our expectation is that datasets such as those that we are producing will result a valuable tool under this respect.

For this reason, we will pursue several directions in our future work to consolidate our achievements. Firstly, we plan to consolidate the dataset, by running several data collection campaigns over different periods of the year and in different weather conditions. Secondly, we will consolidate the integration of the prototype with fog/edge and cloud technologies to explore more complex applicative scenarios which may involve both IoT devices and multimedia communications, and, finally, we will consolidate the collection of weather forecast data from public channels to complement the dataset.

Acknowledgement

This work has been partly funded by the Spanish Ministry of Economy and Competitiveness under projects PLATINO (TEC2017-86722-C4-4-R) and CitiSim Itea3 (TSI-102107-2016-8 ITEA3 Num. 15018) and by the Regional Government of Castilla-La Mancha under project SymbIoT (SBPLY/17/180501/000334). Xavier del Toro García has received financial support from the European Regional Development Fund (Fondo Europeo de Desarrollo Regional, FEDER).

References

[1] A. Kansal and M. B. Srivastava, "An environmental energy harvesting framework for sensor networks," in Proceedings of the International Symposium on Low Power Electronics and Design, ser. ISLPED '03. New York, NY, USA: ACM, 2003, pp. 481–486.

[2] F. A. Kraemer, D. Ammar, A. E. Braten, N. Tamkittikhun, and D. Palma, "Solar Energy Prediction for Constrained IoT Nodes Based on Public Weather Forecasts," in Proceedings of the Seventh International Conference on the Internet of Things. New York, NY, USA: ACM, 2017, pp. 2:1–2:8.

[3] A. Kansal, J. Hsu, S. Zahedi, and M. B., Srivastava, "Power Management in Energy Harvesting Sensor Networks", ACM Trans. Embed. Comput. Syst.,vol. 6, no. 4, Sep. 2007.

[4] C. Moser, J. Chen, and L. Thiele, "Dynamic power management in environmentally powered systems," in 15th Asia and South Pacific Design Automation Conference (ASP-DAC), Jan 2010, pp. 81–88.

[5] S. Escolar, S. Chessa, and J. Carretero, "Optimization of Quality of Service in Wireless Sensor Networks Powered by Solar Cells," in IEEE 10th International Symposium on Parallel and Distributed Processing with Applications, July 2012, pp. 269–276.

[6] S. Escolar, S. Chessa, and J. Carretero, "Energy management in solar cells powered wireless sensor networks for quality of service optimization," Personal and Ubiquitous Computing, vol. 18, no. 2, pp. 449–464, 2014.

[7] S. Escolar, S. Chessa, and J. Carretero, "Energy management of networked, solar cells powered, wireless sensors," in Proceedings of the 16th ACM international conference on Modeling, analysis & simulation of wireless and mobile systems. ACM, 2013, pp. 263–266.

[8] S. Escolar, S. Chessa, and J. Carretero, "Energy-neutral networked wireless sensors," Simulation Modelling Practice and Theory, vol. 43, pp. 1–15, 2014.

[9] S. Escolar, S. Chessa, and J. Carretero, "Quality of service optimization in solar cells-based energy harvesting wireless sensor networks,", Energy Efficiency, pp. 1–27, 2016.

[10] A. Caruso, S. Chessa, S. Escolar, X. del Toro, and J. C. López, "A Dynamic Programming Algorithm for High-Level Task Scheduling in Energy Harvesting IoT,", IEEE Internet of Things Journal, vol. 5, no. 3, pp. 2234–2248, June 2018.

[11] S. Escolar, A. Caruso, S. Chessa, X. del Toro, F. J. Villanueva, and J. C. López, "Statistical Energy Neutrality in IoT Hybrid Energy-Harvesting Networks," in IEEE Symposium on Computers and Communications (ISCC). IEEE, June 2018, pp. 444–449.

[12] N. Sharma, J. Gummeson, D. Irwin, and P. Shenoy, "Cloudy computing: Leveraging weather forecasts in energy harvesting sensor systems," in 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, June 2010, pp. 1–9.

[13] A. Cammarano, C. Petrioli, and D. Spenza, "Online energy harvesting prediction in environmentally powered wireless sensor networks," IEEE Sensors Journal, vol. 16, no. 17, pp. 6793–6804, Sep. 2016.

[14] M. Kuzman, X. del Toro, S. Escolar, A. Caruso, S. Chessa, and J. C. López, "A testbed and an experimental public dataset for energy-harvested iot solutions," in 17th IEEE International Conference on Industrial Informatics (INDIN), Helsinki-Espoo, Finland, Jul 2019.

[15] A. Caruso, S. Chessa, S. Escolar, X. del Toro, M. Kuzman, and J. C. López, "Experimenting forecasting models for solar energy harvesting devices for large smart cities deployments," in IEEE ISCC 2019 - Workshop on Management Of Cloud and Smart City Systems (MoCS) (IEEE ISCC WKSHPS-MoCS 2019), Barcelona, Spain, Jun. 2019.

[16] S. Escolar, X. del Toro, F. J. Villanueva, M. J. Santofimia, D. Villa, F. Rincón, J. Barba, and J. C. López, "The PLATINO Experience: A LoRa-based Network of Energy-Harvesting Devices for Smart Farming," in 34th Conference on Design of Circuits and Integrated Systems

(DCIS), Bilbao, Spain, Nov 2019.
[17] C. Puliafito, E. Mingozzi, F. Longo, A. Puliafito, and O. Rana, "Fog computing for the internet of things: A survey," ACM Trans. Internet Technol., vol. 19, no. 2, pp. 18:1–18:41, Apr. 2019. [Online]. Available: http://doi.acm.org/10.1145/3301443
[18] H. F. Atlam, R. J. Walters, and G. B. Wills, "Fog computing and the internet of things: A review," Big Data and Cognitive Computing, vol. 2, no. 2, 2018. [Online]. Available: https://www.mdpi.com/2504-2289/2/2/10

Stefano Chessa received the PhD degree in Computer Science from the University of Pisa in 1999, and currently he is Associate Professor at the Department of Computer Science of the University of Pisa, vice-chair of the BSc and MSc curricula in "Computer Science" of the University of Pisa and also Research Associate of the Information Science and Technology Institute of the CNR. He has worked several EU projects, in particular he had been site leader of the EU FP7 projects RUBICON and DOREMI. He has co-authored more than 150 papers published on international journals and conference proceedings, and he has been member of several program committees of international conferences. His current research interests are in the areas of internet of things, smart environments and wireless communications in general.

Soledad Escolar received her MSc and PhD degrees from University Carlos III of Madrid, Spain, in 2004 and 2010. During the period between 2004 and 2014 she has been a teaching assistant and researcher in the Computer Science Department at the same university. Currently she is an Assistant Professor in School of Computer Science at University of Castilla-La Mancha, Ciudad Real, Spain. Her primary research interests are mainly focused on Wireless Sensor Networks, highlighting energy management algorithms, network protocols, adaptive behavior, Internet of Things and Smart Cities.

Antonio Caruso received the MS degree ("cum laude") in Computer Science from the University of Pisa, Italy, and the PhD in Computer Science from the same University. From 2005 he joined the Mathematicciaoal and Physics Department of the University of Salento as Assistant Professor. He received the Innovation Award from Italian-Canada in 2017. He has been member of several program committees of conferences and workshops, and published on international journals and conference proceedings mainly in the area of: mobile distributed systems, internet of things, smart environment, wireless sensor and ad-hoc networks, underwater networks, and distributed algorithms.

Xavier del Toro received the degrees of Technical Engineer on Industrial Electronics and Engineer in Automatic Control and Industrial Electronics in 1999 and 2002, respectively, from the Universitat Politècnica de Catalunya (Spain). He received the PhD degree from the University of Glamorgan (UK) in 2008. From September 2005 until October 2006 he was a Marie Curie Research Fellow at Politecnico di Bari (Italy). Since 2008 he is working as a researcher in the University of Castilla-La Mancha (Spain). His research interests include power electronics, renewable energies, energy storage and energy harvesting systems.

Juan C. López received his MS and Ph.D. degrees in Telecommunication Engineering from the Technical University of Madrid in 1985 and 1989, respectively. From September 1990 to August 1992, he was a Visiting Scientist in the Department of Electrical and Computer Engineering at Carnegie Mellon University, Pittsburgh, PA (USA). His research activities center on embedded system design, distributed computing and advanced communication services. His contributions have been published in journals, book chapters and conference proceedings. He has led different national and international projects in those areas and has served as program committee member, session chair and reviewer in the main international conferences on design automation and computer architecture. From 1989 to 1999, he has been an Associate Professor of the Department of Electrical Engineering at the Technical University of Madrid. Currently, Dr. López is a Professor of Computer Architecture at University of Castilla-La Mancha where he served as Dean of the School of Computer Science from 2000 to 2008, holding now the Indra Chair. He is and has been a member of different panels of the Spanish National Science Foundation and the Spanish Ministry of Education and Science, regarding the Information Technologies research programs.

Deploying APIs: Edge vs Cloud Environments

Sergio Laso[1], *Daniel Flores*[1], *Jose Garcia-Alonso*[1], *Juan Manuel Murillo*[1] *and Javier Berrocal*[1],
[1] *School of Polytechnic, University of Extremadura, Caceres, Spain.*
slasom@unex.es; dfloresm@unex.es; jgaralo@unex.es; juanmamu@unex.es; jberolm@unex.es;

1. Introduction

During the last few years, edge and IoT devices have increased their computing and storage capabilities enormously, being able to not only sense and forward the gather information but also to store and compute it [1]. This increase has also allowed developers to take these devices as possible destination for the deployment of their applications and APIs.

Devices closer to the users can better satisfy important requirements of some APIs and applications, such as responsiveness, location-awareness, etc. [2]. Nevertheless, these devices also present some important restrictions regarding resource consumption (battery, data traffic and so on) that should be taken into account during the decisions process of where to deploy them [1]. Currently, developers have to carefully analyze every deployment layer and the available devices to identify the environment that better meet the system requirements. This is a complicate process in which developers need tools assisting them or they end up selecting the environment in which they feel most comfortable.

In addition, developing and deploying an API or an application on IoT and edge devices is not an easy task. Usually, these devices have stringent requirements and closed operating system. For instance, Android-based devices have some security features hindering how the API should be developed and how other users can get information from it [3]. This is another point leading developers to choose environments for which development is simpler.

In this paper, we propose, first, an application assisting developers in the decision making process of selecting the best possible environment for their APIs; and, second, a set of tools facilitating the generation of the APIs skeleton for the selected environment. Thus, developers can easily evaluate and select any environment for the deployment of their APIs without having to devote an extra effort.

2. Motivation and Related Works

During the last few years, the Cloud has been the predominant environment for the deployment of APIs. In this environment the most demanding features, like storing or computing the managed data, are offloaded providing improved scalability, fault tolerance and a greater control of the operational costs. It also allowed developers to implement applications with good response time that can be visualized and used from end devices, that until a few years ago had very limited capabilities.

In the last few years, edge and IoT devices' capabilities have increased tremendously. Their storage and computational capabilities have increased in order to be able to execute more complex tasks. This allowed researchers to develop new paradigms, in which the whole or part of an application or API is onloaded in edge devices, reducing the network load and the dependency of Cloud environments and improving the response time. For instance, by using data closer to, or even inside, the targeted device.

Therefore, developers not only have to take decisions about how to develop a specific API but also where it should be deployed to better meet the system requirements. This is complex trade-off among the system requirements, and the capabilities and limitations of the different devices in the environment. Deploying and API on the Cloud could increase the system scalability or the fault tolerance, but could also negatively affect the operational cost, location-awareness and responsiveness [2]. Deploying an API on edge or end devices could positively impact these requirements, but they also present some important limitations. For instance, a constrained factor of these devices is the resource consumption (battery, data traffic, etc.) [4]. Many of these devices, such as mobile phones, smart speakers, etc. are battery powered. Likewise, some of them have to interact using the mobile network, either because they are mobile or because of their specific situation, which entails a consumption of the data plan. In the deployment of any API, the correct management of these resources is crucial for the user satisfaction. It is well known that resource consumption, is a factor determining the application success [5]. All these dimensions have to be taken into account to select the deployment architecture that best fit the system requirements, doing a trade-off among the capabilities and limitations of every involved device and the system requirements.

Some works both in the academia and in the industry domains are focused on analysing the final developed and

deployed application for identifying if those requirements are met. For instance, Giovani et al. [6] presented some years ago a monitoring framework able to adjust the resources assigned to the deployed application or to face transient failures replicating and restarting components to provide resiliency. Leitner et al. [7] proposed a tool to continually re-evaluate the cost of an application in the background. Or, for Android devices, Batterystats [8] could be used to collects battery data about the consumption of an specific application. Other proposal, however, are focused on evaluating during the early stages of the development process how these requirements can be met, helping in the decision making process of which deployment architecture is the most suitable. For instance, [9] proposes different heuristic to identify where microservices should be deployed to better meet some requirements such as responsiveness. Likewise, the authors of this paper have been working on defining a conceptual framework [1] to identify in early development stages what deployment architecture is the most appropriate to reduce the resource consumption. However, the application of the previous framework remains a manual process. The developer must specify the application, apply the established formulas and try to identify the consumption trends of the application for each architecture. For small applications, this is simple, but as the application grows, the application of this framework becomes difficult. Tools assisting in this process are required to easily selected the best deployment architecture for each application.

Once identified the most appropriate deployment architecture for achieving the system requirements, the API has to be developed and deployed. Currently, the specification and development of microservices is supported by a large number of tools that facilitate the work of the developer. Specifications such as OpenAPI [10], are widely used to detail an API, generate the documentation and, even, perform tests. This type of tools can even generate the source code for different technologies such as Node.JS, Kotlin, JAX-RS, etc. [11], but they are focused on deploying the API on a server or on a cloud environment. There are fewer commercial tools supporting other deployment architectures, based on Fog, Mist or Edge computing. However, at the research level, more and more proposals are presented detailing how to use these specifications to generate the skeleton and to deploy microservices on other elements of the network. For instance, Noura et al. propose WoTDL2API tool, that automatically generates a running RESTful API based on the OpenAPI specification and its code generation toolchain to control different IoT devices [12]. The generated source code can be deployed on edge devices, such as gateways, routers, and so on. Nevertheless, tools are still required to generate APIs that can be deployed on IoT devices.

In the next two sections, we present, first, a tool assisting developers in the decision making process of selecting the best deployment architecture for an application; and, second, a set of tool that can be used to generate the skeleton of an API that can be deployed on a Cloud environment or in Android-based IoT devices (such as smartphones).

3. Tools helping in the decision-making process

To assist developers in selecting the best deployment architecture, a web application [13] has been created for estimating the resource consumption for different deployment architecture options (such as based on the Cloud or on IoT devices), as well as its evolution for different scenarios.

In order to generate these specifications, the developed application requires the specification of the most important features of the API for the different deployment architectures that must be estimated. This specification is based on a set of common primitive operations together with their associated resource consumptions identified in the conceptual framework presented in [1]. Any application can be described as a combination of these primitives, thereby allowing the application's resource consumption to be estimated for each architectural option.

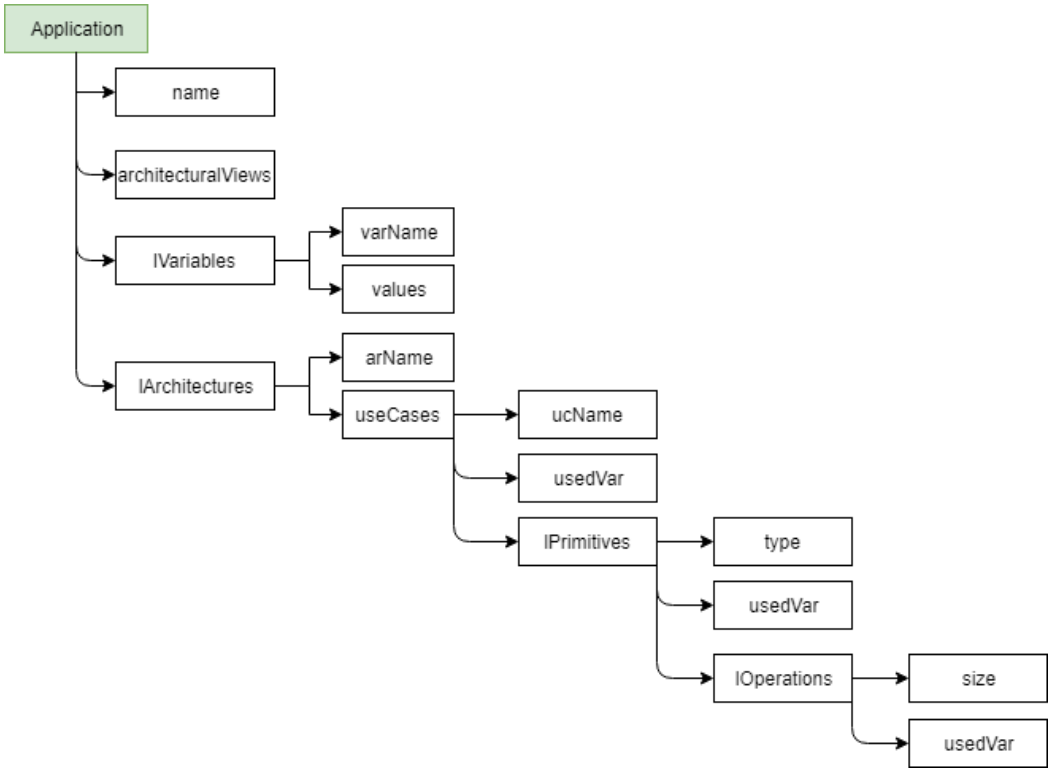


Figure 1. Scheme of the application specification to estimate consumption.

The specification of the application to estimate is provided in a JSON format file. Fig.1 shows the scheme of said file. The most important elements are:

- *name*, name of the application.
- *architecturealViews*, list of the deployment architectural options that have to be estimated.
- *lVariables*: list of variables and parameters that can be used to simulate the behavior and evolution of the application (such as the number of users, the size of the information they transmit, etc.).
- *lArchitectures*: specification of the application behavior for each architectural option. First, the name of the architectural option (*arName*) has to be specified and, second, the behavior of the use cases or features of the application (*useCases*) has to be detailed. Each feature is detailed using the primitive operations that can simulate the desired behavior. The information that has to be specified is:
    1. The name of the use case (*ucName*).
    2. If there is any variable affecting the use case and its behavior (*usedVar*).
    3. And, the list of primitives in which the use case is broken down (*lPrimitives*). For each primitive the developer has to detail: *type*, the primitive operation by its name; *usedVar*, the variables affecting the behavior of the functionality; and, *lOperations*, which are the parameters that these primitives expect to receive in order to better estimate their exact consumption (such as the size of the information to be received or sent).

Once the different architectural options are detailed and provided to the web application, it returns the estimated consumption of the application in CSV format, so it can be easily processed to visualize the consumption of each deployment architectural option. Fig.2. shows and excerpt of the CSV file and a chart showing the estimated consumption of a Cloud-based deployment architecture.

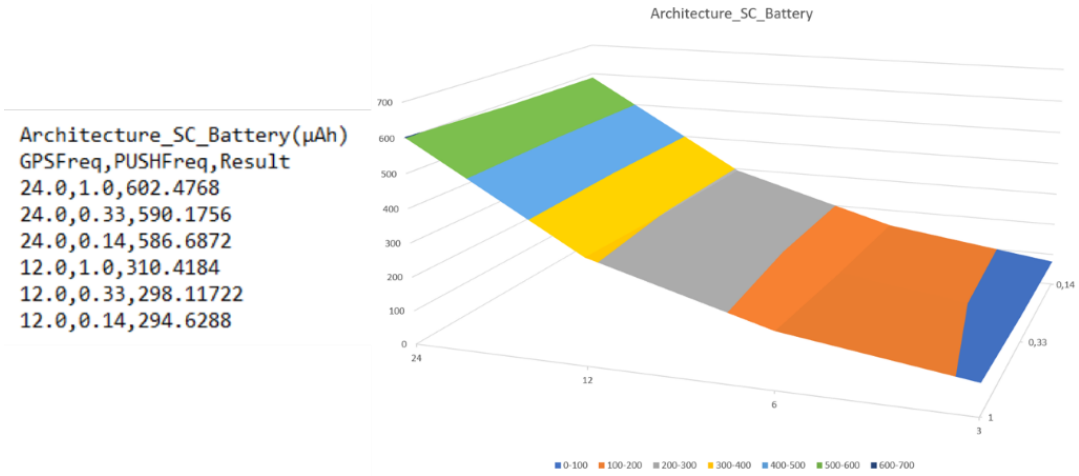


Figure 2. Example of results and visualization.

This tool assists developers on identifying what deployment environment can better meet the system requirements and when and application or an API has to be migrated from one environment to another in order to keep meeting the system requirements.

4. Facilitating the generation of applications for different deployment architectures

Likewise, the authors of this paper have recently proposed a set of tools [14] to generate and deploy a running RESTful API based on the OpenAPI standard on Android-based end devices such as smartphones, IoT devices, etc., simplifying the development of these applications.

As can be seen in Fig.3, the first step is to define the application, this task is done using OAI (OpenAPI Specification) [15]. The different features and microservices of the API has to be specified as if they were going to be deployed on a Cloud environment.

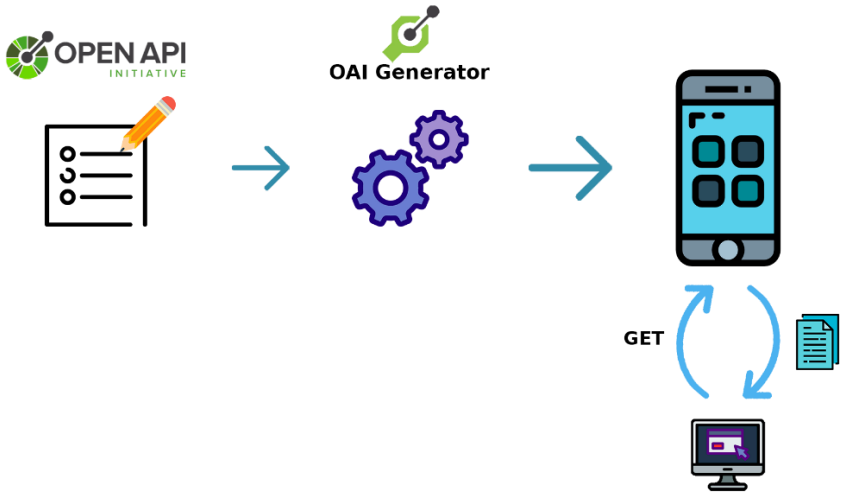


Figure 3. Process to Generate Client-Centric Application

Second, the API skeleton is generated. Currently, there is a OpenAPI Generator that can be used to generate the scaffolding of an API that is going to be deployed on a Cloud environment. For developers needing to deploy the API in an end device, all this source code has to be manually written hindering its development. In order to facilitate the selection of this deployment architecture, the OpenAPI Source Code Generator has been extended so that services or functionalities can be automatically generated and deployed on Android-based IoT devices. Then, these services can then be consumed by third entities.

As Android-based devices (mainly smartphones) have the communication channels restricted in order to increase the system's security, the communication logic of an API Rest for this devices has been implemented (and provided as

part of the generated API skeleton) using Firebase Cloud Messaging (FCM) [16].

Finally, any microservice or functionality can be invoked sending a Push Notification to the IoT devices. Thus, the functionality is executed in the end devices and the result is returned to the client.

This set of tools allows developers to easily deploy APIs on Android-based IoT devices using the same technologies that they usually use when they have to be deployed on Cloud environment reducing the technological gap between both environments.

5. Conclusions

APIs and applications' backend until now have been designed and developed to be deployed on a Cloud environment because of its computing capabilities, fault tolerance, responsiveness, etc. Nevertheless, the ever-increasing capabilities of other elements in the networks has fostered the deployment of these APIs and application on these devices to better meet some requirements such as scalability, interoperability, real-time responsiveness, security or location-awareness.

Nevertheless, for developers to apply the best deployment architecture, first, they need tools helping in the decision making process by providing information for each option about the degree of compliance of the requirements; and, secondly, tools supporting and reducing the effort required to develop and deploy the API for the selected architecture are needed. If these tools are not provided developer tend to use only known architectures and those with a greater support.

In this paper, we presented some tools assisting developers in both steps of the development process, reducing the technological gap between deploying an application on a Cloud environment or on edge or IoT environment.

Acknowledgement

This work was supported by 4IE project (0045-4IE-4-P) and 4IE+ project (0499_4IE_PLUS_4_E) funded by the Interreg V-A España-Portugal (POCTEP) 2014-2020 program, by the Spanish Ministry of Science, Innovation and Universities (RTI2018-094591-B-I00), by the Department of Economy and Infrastructure of the Government of Extremadura (GR18112, IB18030), and by the European Regional Development Fund.

References

[1] J. Berrocal *et al.*, «Early analysis of resource consumption patterns in mobile applications», *Pervasive Mob. Comput.*, vol. 35, pp. 32-50, feb. 2017.

[2] P. Bellavista, J. Berrocal, A. Corradi, S. K. Das, L. Foschini, y A. Zanni, «A survey on fog computing for the Internet of Things», *Pervasive Mob. Comput.*, vol. 52, pp. 71-99, ene. 2019.

[3] H.-H. Cho y J.-B. Kim, «A Study on the Security Vulnerability for Android Operating System», *Proc. Korean Inst. Inf. Commucation Sci. Conf.*, pp. 224-226, 2015.

[4] J. Ren, H. Guo, C. Xu, y Y. Zhang, «Serving at the Edge: A Scalable IoT Architecture Based on Transparent Computing», *IEEE Netw.*, vol. 31, n.º 5, pp. 96-105, 2017.

[5] H. Qian y D. Andresen, «Extending Mobile Device's Battery Life by Offloading Computation to Cloud», en *Proceedings of the Second ACM International Conference on Mobile Software Engineering and Systems*, Piscataway, NJ, USA, 2015, pp. 150–151.

[6] G. Toffetti, S. Brunner, M. Blöchlinger, F. Dudouet, y A. Edmonds, «An Architecture for Self-managing Microservices», en *Proceedings of the 1st International Workshop on Automated Incident Management in Cloud*, New York, NY, USA, 2015, pp. 19–24.

[7] P. Leitner, J. Cito, y E. Stöckli, «Modelling and Managing Deployment Costs of Microservice-based Cloud Applications», en *Proceedings of the 9th International Conference on Utility and Cloud Computing*, New York, NY, USA, 2016, pp. 165–174.

[8] «Profile battery usage with Batterystats and Battery Historian», *Android Developers*. [On line]. Disponible en: https://developer.android.com/studio/profile/battery-historian. [Accessed: 25-ago-2019].

[9] M. Selimi, L. Cerdà-Alabern, M. Sánchez-Artigas, F. Freitag, y L. Veiga, «Practical Service Placement Approach for Microservices Architecture», en *2017 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, 2017, pp. 401-410.

[10] *The OpenAPI Specification Repository. Contribute to OAI/OpenAPI-Specification development by creating an account on GitHub*. OpenAPI Initiative, 2019.

[11] *OpenAPI Generator allows generation of API client libraries (SDK generation), server stubs, documentation and configuration automatically given an OpenAPI Spec (v2, v3): OpenAPITools/openapi-gener.*. OpenAPI Tools, 2019.

[12] M. Noura, S. Heil, y M. Gaedke, «Webifying Heterogenous Internet of Things Devices», en *Web Engineering*, 2019, pp. 509-513.

[13] J. Berrocal y S. Laso, «Resource Consumtion Estimation Tool». [On line]. Disponible en: https://api-consumptions.herokuapp.com/swagger-ui.html. [Accessed: 25-ago-2019].

[14] J. Berrocal y S. Laso, «OpenAPI Android Generator». [On line]. Disponible en: https://openapi-generator-spilab.herokuapp.com/swagger-ui.html. [Accessed: 25-ago-2019].

[15] «OpenAPI Specification | Swagger». [On line]. Disponible en: https://swagger.io/specification/. [Accessed: 25-ago-2019].

[16] «Firebase Cloud Messaging», *Firebase*. [On line]. Disponible en: https://firebase.google.com/docs/cloud-messaging. [Accessed: 25-ago-2019].

Sergio Laso is a researcher and MSc student in Computer Science Engineering at the University of Extremadura (Spain). He is currently working at the Computing and Telematics Systems Department. His research interests are mobile computing, pervasive systems, context-awareness, fog computing and the Internet of Things.

Daniel Flores-Martín is a PhD. student at the University of Extremadura (Spain). He is currently working at the Computing and Telematics Systems Department of the University of Extremadura. His research interests are mobile computing, context-awareness, pervasive systems, crowd sensing and Internet of Things. Flores-Martin received an MSc in Computer Science Engineering from the University of Extremadura.

Jose Garcia-Alonso is a co-founder of Gloin and a PhD candidate and Associate professor of software engineering at the University of Extremadura. His research interests include software product lines, software architectures, model-driven development, and framework development. Garcia-Alonso received an MSc in software engineering from the University of Extremadura.

Juan Manuel Murillo is a co-founder of Gloin and a full professor at the University of Extremadura. His research interests include software architectures, mobile computing, and cloud computing. Murillo has a PhD in computer science from the University of Extremadura.

Javier Berrocal received the Ph.D. degree in computer science from the University of Extremadura, Spain, in 2014. In 2016, he obtained an Associate position at the University of Extremadura. His main research interests are mobile computing, context awareness, pervasive systems, crowd sensing, the Internet of Things, and fog computing. He is a cofounder of the company Gloin, which is a software-consulting company.

Human Mobility-based Deployment of Edge Data Centers in Urban Environments

Piergiorgio Vitello[1], *Andrea Capponi*[1], *Claudio Fiandrino*[2], *Guido Cantelmo*[3] *and Dzmitry Kliazovichz*[4],

[1]*FSTC/CSC, University of Luxembourg, Luxembourg.*
[2]*IMDEA Networks Institute, Madrid, Spain.*
[3]*Technical University of Munich (TUM), Germany.*[4]*ExaMotive, Luxembourg.*
piergiorgio.vitello@uni.lu; andrea.capponi@uni.lu; claudio.fiandrino@imdea.org; g.cantelmo@tum.de; kliazovich@ieee.org;

Abstract
Multi-access Edge Computing (MEC) brings storage and computational capabilities at the edge of the network into so-called Edge Data Centers (EDCs) to improve the performance of low-latency applications. For this purpose, effective deployment of EDCs in urban environments is crucial to minimize outages and perform load balancing properly. Our study specifically tackles this issue. To deeply understand the variation of computational demand of EDCs at urban scale, the analysis of cities complex dynamics assumes a primary role. This work aims to develop a methodology for an effective deployment of MEC EDCs in urban environments by considering the mobility of citizens and their spatial patterns. To this end, we propose and compare two heuristics. In particular, we present the mobility-aware deployment algorithm (MDA) that outperforms approaches that do not consider citizens mobility. Simulations are conducted in Luxembourg City by extending the CrowdSenSim simulator and show that efficient EDCs placement significantly reduces outages.

1. Introduction

The fifth generation (5G) mobile networks rely on Software-Defined Networking (SDN) and Network Function Virtualization (NFV) to support next generation services. Radio access and core functions are virtualized and executed in edge data centers (EDCs) according to the Multi-access Edge Computing (MEC) principle. MEC is a key enabler for 5G mobile networks and was standardized by the European Telecommunications Standards Institute (ETSI) [1]. Formerly known as Mobile Edge Computing, MEC aims at providing computing service closer to the end user by bringing applications and services at a close distance to the end user [2]. Thus, it finds applicability in scenarios where locality and low-latency are essential [3].

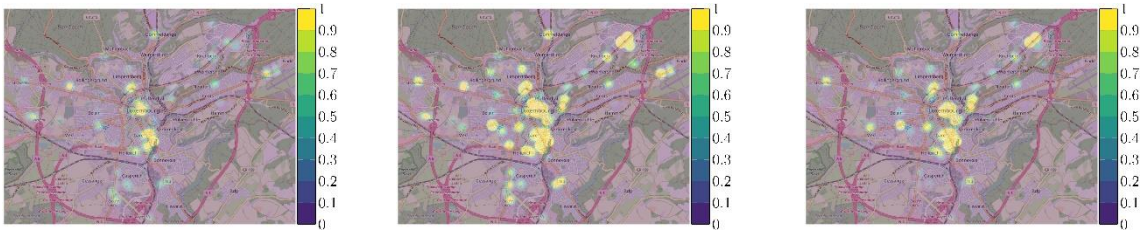
The edge, also known as MEC host, is a data center or nano data center deployed close to the base stations inside an infrastructure owned by a mobile network operator (MNO). The edge provides computing functionalities and can aggregate virtualized core and radio network functions of the mobile network. Such principle originates from the concept of EDCs and enables resource-constrained mobile devices to prolong battery lifetime while enhancing and augment performance of the mobile applications [4].

To this date, the research of edge computing has mainly focused on resource management and allocation by trading power consumption and communication delays while seminal works have mainly focused on the definition of architectural design principles [5]. Emulation platforms for research in the area have only started to appear recently [6] and little attention has been paid to the problem of resource deployment.

EDC deployment is a particularly interesting and challenging problem in the context of smart cities. To the best of our knowledge, only a vision paper has explored such area by assessing the feasibility of leveraging three different infrastructures, i.e., cellular base stations, routers, and street lamps and analyze the potential city coverage if only a subset of these elements is upgraded to furnish EDC capabilities [7]. This study is an important step forward to solve the problems of coverage, EDC selection and user-to-EDC assignment. However, it does not consider that urban mobility plays a key role in designing architectures for Edge computing. Citizens mobility is influenced by many factors, including trip purposes and geographically imposed restrictions, such as home and work location. These complex phenomena (called urban dynamics in the rest of the paper) consistently impact the computational demand of BSs and EDCs that changes over time and in turns devise effective EDC deployments.

In this paper, we bring the research in edge computing one step forward. Specifically, we tackle the problem of EDCs deployment in a smart city context by considering two factors. First, we consider cellular connectivity for network access and assume that EDCs should only be deployed at current Base Stations (BSs) sites to re-use already deployed

infrastructure (e.g., power supply, cabinets on roofs). Thus, our solution is capital-expenditure free for mobile network operators. Second, we focus on human mobility. Within a city, complex dynamics regulate the inter-dependency of land use and citizens movements [8], i.e., the spatial distribution of citizens and locations they visit that determine mobility patterns. Similarly to [9], we exploit crowdsensed data to infer human mobility with the goal of determining estimates of computing demand and the optimal EDC deployment that minimizes outages. Specifically, we leverage Google Popular Times to estimate citizens mobility that reflects daily urban patterns. For these reasons, we focus on citizens mobility during a day and consider LTE traffic generated from mobile users.



d) H: 8.00 **e) H: 12.00** **f) H: 17.00**

Fig. 1 Traffic generation in Luxembourg City at different hours of a working day

Fig. 1 shows the heatmaps of the potential computing demand of the BSs in Luxembourg City according to the citizens mobility based on Google Popular Times (https://support.google.com/business/answer/2721884) in a weekday. BSs under heavy loads are mostly around the railway stations when citizens commute (i.e., H: 8:00 and H: 17:00) and sparse around the city during the day. By considering different hours of a day, loads of BSs vary and, in turns, the potential computational demand at EDCs, which motivates our research in EDC deployment.

2. MEC EDCs Deployment based on Human Mobility

This Section formulates the problem of EDC distribution over urban environments and presents the system model that captures MEC dynamics (both at computing and networking level) and citizens mobility.

2.1 MEC Model

Given a set of BSs characterized by latitude and longitude that define their location in the city, our aim is to deploy a set of EDCs at BS sites to re-use already deployed infrastructure (e.g., power supply, cabinets on roofs). This provides full spatial coverage and is more cost-effective than creating new EDC sites [7]. We consider as Key Performance Indicator (KPI) the latency outage probability of the system O, defined as the probability that the latency the user observes and is over the maximum acceptable delay bound defined in the form of SLA agreement for the current application. We need for the latency outage probability of the system to measure the Round-Trip-Time (RTT) to capture the fact that if a user does not receive a reply from the EDC, the task is not accomplished and O increases. This can happen when the EDC rejects the incoming task because it is overloaded, or when the user does not receive the reply in due time because of either processing or networking delays.

Given a fixed number of EDCs to deploy, our problem consists in finding a match with the location of existing BSs to minimize the average latency outage probability of the entire system (e.g., to maximize the computational capacity of the system).

To assign EDCs to BSs, the city environment is divided into a set of regions. The EDC of each region is connected to all BSs within the region and responsible for both applications and baseband processing. Each EDC has a fixed number of servers with an equal service rate μ. When the service rate is not sufficient to fulfill a task in due time, it is rejected by the system. We assume no service migration across different EDCs when the users move within the same region. Otherwise, the service is simply detached from an EDC and attached to another according to, e.g., a micro-services stateless paradigm. Finally, we assume the mobile users are always connected to the closest BS. Users generate heterogeneous application-dependent types of tasks [10]. The task arrival is modeled with a Poisson process with arrival rate λ for each user. To access the EDC processing, a user sends a message which is acknowledged if EDC resources permit its execution. The latency L of the request/reply exchange includes both network and processing delays [11]. The network delay Dp consists of different components, such as transmission, propagation, queuing, and routing. The processing delay Dc depends on application processing and packet processing at the network level. Each EDC is modeled as a M/M/Ns queue with Ns servers. The processing delay, which represents the task execution time

in an EDC, is calculated considering the fraction of accepted tasks in the EDC and the average queueing time, which is given by the Erlang's Formula [12].

Users mobility defines spatial patterns of citizens movements and their social interactions, influencing the demand for computing resources. In this work, to characterize urban mobility, we exploit the popularity of Local Businesses (LB) taken from Google Popular Times, given in per-hour values normalized between the weekly maximum and minimum number of customers of each LB. Note that the real number of customers remains unknown. Hence, one of our contributions explained hereafter, is a new approach to estimate the number of customers from the coarse measurements available. The popularity metric is then used to approximate users temporal distribution among different LBs. For each type of LB, we consider a random value between 0 and Nmax, where Nmax is the maximum value of customers. This value and the average waiting time of staying in a LB permits to compute the number of people who remained at that LB. The aggregation of different LBs in a region defines how crowded a district is.

2.2 EDCs Deployment Policies

The placement policy and BSs assignment to EDCs are keys for effective EDCs deployment. We propose and compare two placement policies. The first one is called distributed deployment algorithm (DDA) and deploys EDCs so that they are the centroids of a cluster composed of a set of BSs that all share a similar distance. The second policy, called mobility-aware deployment algorithm (MDA), considers the mobility of citizens and their social interactions in urban environments to calculate the expected computational demand and distribute edge resources by exploiting them as weights.

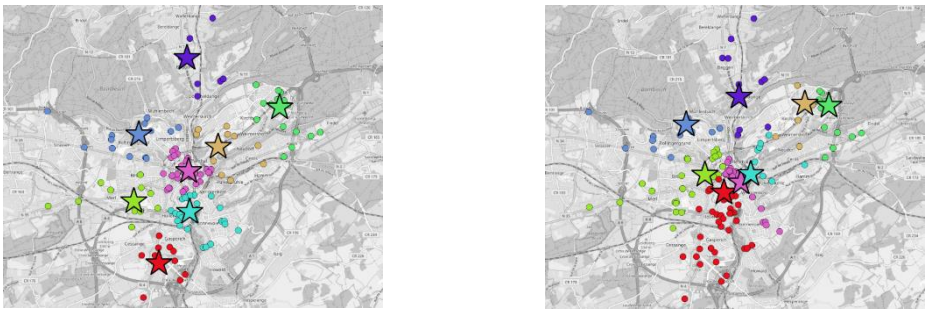
DDA places EDCs and assigns BSs to them by exploiting the k-medoids clustering algorithm, which is similar to the more famous k-means but chooses the centroids between the input data. In other words, it clusters BSs and chooses EDCs as centers of the clusters between the BSs within a cluster. The main shortcomings of this approach consist in some under-utilized EDCs and others that suffer of big delays due to overloads of computational demand leading to high values of outage probability. To overcome these issues, two possible directions can be investigated. First, to propose a more effective placement of EDCs among the BSs. Second, to allocate servers among EDCs proportionally with the computational demand.

MDA aims to decrease the overall outage probability of the system by considering where the computational demand is higher according to the spatial patterns of citizens. In other words, it is based on the idea to consider the complex dynamics of a city (e.g., user mobility and social interactions) to propose a more effective placement of edge resources. As DDA, MDA places EDCs among BSs by exploiting the k-medoids algorithm, but it assigns EDCs among BSs by computing a cost based on the number of requests received by BSs and corresponding computational demand for EDCs.

3. Performance Evaluation

To conduct simulations, we consider the real infrastructure of 141 BSs in Luxembourg City for public mobile communication network over 50W (https://data.public.lu/fr/datasets/cadastre-gsm/), which is imported as a layer on a map to extract latitude and longitude of each BS (https://map.geoportail.lu/). Users generate traffic with an arrival rate λ_i set in the range [0 - 2.99] [10]. The service rate of each server is μ=100.

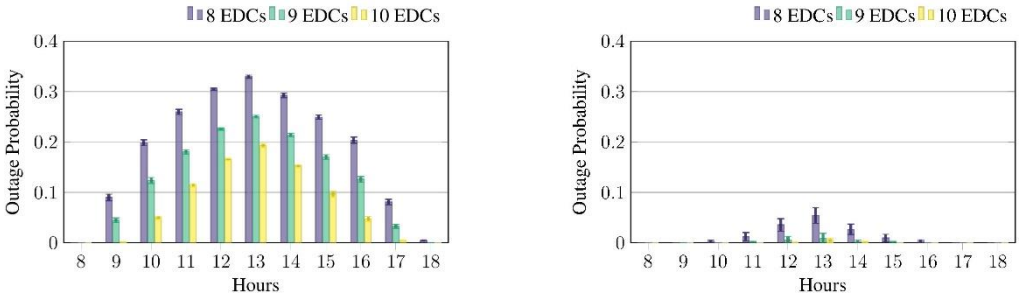
To simulate user mobility in realistic urban environments, we extend CrowdSenSim [13] originally developed for mobile crowdsensing [14]. We operate in Luxembourg City and consider 1,083 LBs belonging to 13 different categories (e.g., restaurants, banks, etc.) from Google Popular Times. 100,000 pedestrian walk on the city street network over the city according to a mobility model weighted through the popularity of LBs. The simulation period is 24 hours of a working day given by the average of days between Monday and Friday.



a) DDA **b) MDA**

Fig. 2 Distributed (DDA) and Mobility-aware (MDA) Deployment Algorithms

Fig. 2 presents the deployment of 8 EDCs with DDA and MDA approaches in Luxembourg City. Circles and stars represent BSs and EDCs, respectively. BSs are assigned to an EDC of the same color. This result unveils that considering the mobility of citizens leads to a significantly different EDC deployment. On the one hand, DDA deploys EDCs so that all the controlled BSs experience a similar distance. On the other side, the MDA approach deploys EDCs among BSs that experience higher computational demands. Specifically, with MDA most of the EDCs tend to be deployed closer to the city center and two of them in the north-eastern district of the city (Kirchberg area), which are the most important working and business districts of Luxembourg City and are very crowded during working days, especially at lunchtimes.



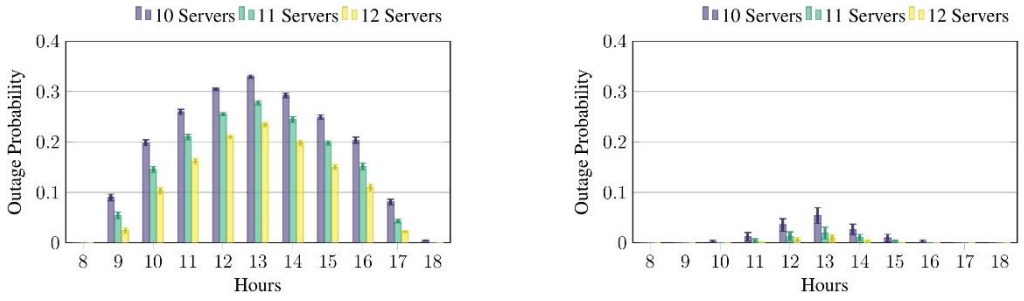
a) DDA **b) MDA**

Fig. 3 Total Outage Probability in a working day with a different number of EDCs (number of servers per EDC fixed to 10)

Fig. 3 shows the per-hour outage probability in a working day for the proposed approaches with a fixed number of servers per EDC (10) and varying the number of EDCs.

Fig. 3.a illustrates the outage probability for the DDA approach. MDA clearly outperforms DDA, as shown in Fig. 3.b. Interestingly, the results show that the variation of the number of EDCs shows a different behavior of the two approaches. By increasing the number of EDCs in the city makes the outage probability of decreasing proportionally for DDA. This is not true in MDA as having 9 or 10 EDCs makes little difference.

We now fix the number of deployed EDCs in the city (8) and investigate in Fig.4 the per-hour outage probability by varying the number of servers for each EDC. Fig. 4.a illustrates the DDA approach. The increase of the number of servers per EDC does not decrease the outage probability as it does the deployment of additional EDCs (see result discussed in Fig. 3.a). With a fixed number of EDCs, MDA still outperforms DDA, as Fig. 4.b shows



a) DDA **b) MDA**

Fig. 4 Total Outage Probability in a working day with a different number of servers (number of EDCs in the city fixed to 8)

4. Conclusions

This work tackles the problem of EDCs deployment in urban environments. We show that the performance of low-latency and high-bandwidth applications can improve by considering citizens mobility and their social interactions. We model the computational demand and citizens mobility aiming to formulate a problem for minimizing the outage probability. Then, we propose two heuristic algorithms. The first one (DDA) deploys EDCs on the sole basis of the spatial distances EDCs-BSs, while the second (MDA) is aware of citizens mobility and the expected computational demand. The results show that the policy MDA makes lower the outages, thereby proving that considering citizens mobility for EDCs deployment in urban environments is effective.

References

[19] F. Giust, G. Verin et al., "MEC deployments in 4G and evolution towards 5G," Feb 2018, ETSI White Paper.
[20] T. Taleb, K. Samdanis et al., "On multi-access edge computing: A survey of the emerging 5G network edge cloud architecture and orchestration," IEEE Communications Surveys Tutorials , vol. 19, no. 3, pp. 1657–1681, Third Quarter 2017.
[21] M. Satyanarayanan, "The emergence of edge computing," Computer, vol. 50, no. 1, pp. 30–39, Jan 2017.
[22] P. Mach and Z. Becvar, "Mobile Edge Computing: A Survey on Architecture and Computation Offloading," in IEEE Communications Surveys & Tutorials, vol. 19, no. 3, pp. 1628-1656, thirdquarter 2017.
[23] C. Mouradian, D. Naboulsi et al., "A comprehensive survey on fog computing: State-of-the-art and research challenges," IEEE Communications Surveys Tutorials, vol. 20, no. 1, pp. 416–464, First Quarter 2018.
[24] C. Andrés Ramiro, C. Fiandrino et al., "openLEON: An end-to-end emulator from the edge data center to the mobile users," in Proc. of ACM WiNTECH, 2018, pp. 19–27.
[25] J. Gedeon, J. Krisztinkovics et al., "A multi-cloudlet infrastructure for future smart cities: An empirical study," in Proc. of ACM EdgeSys, Jun 2018, pp. 19–24.
[26] K. Jayarajah, A. Tan et al., "Understanding the interdependency of land use and mobility for urban planning," in Proc. of ACM UbiComp, 2018, pp. 1079–1087.
[27] Y. Zhou, B. P. L. Lau et al., "Understanding urban human mobility through crowdsensed data," IEEE Communications Magazine, vol. 56, no. 11, pp. 52–59, Nov 2018.
[28] M. Jia, J. Cao et al., "Optimal cloudlet placement and user to cloudlet allocation in wireless metropolitan area networks," IEEE Transactions on Cloud Computing, vol. 5, no. 4, pp. 725–737, Oct 2017.
[29] M. C. Filippou, D. Sabella et al., "Flexible MEC service consumption through edge host zoning in 5G networks," CoRR, 2019.
[30] L. Kleinrock, Queueing systems, Volume 1: Theory. Hoboken, NJ, US, 1975.
[31] C. Fiandrino, A. Capponi, et al., "CrowdSenSim: a Simulation Platform for Mobile Crowdsensing in Realistic Urban Environments," in IEEE Access, vol. 5, pp. 3490-3503, 2017.
[32] A. Capponi, C. Fiandrino et al., "A survey on mobile crowdsensing systems: Challenges, solutions and opportunities," IEEE Communications Surveys Tutorials, pp. 1–49, Apr 2019.

Piergiorgio Vitello is a Ph.D. student at the University of Luxembourg. He received the Bachelor Degree and the Master Degree in Software Engineering both from Politecnico di Torino (Italy). His research interests include mobile crowdsensing, urban computing and smart mobility.

Andrea Capponi is a Ph.D. candidate at the University of Luxembourg. He received the Bachelor Degree in Telecommunication Engineering and the Master Degree in Telecommunication Engineering both from the University of Pisa, Italy. He is member of IEEE and ACM and served as Workshop Co-Chair of MoCS 2019. His primary research interests are in the field of mobile crowdsensing, Internet of Things (IoT), and smart cities.

Claudio Fiandrino joined as a postdoctoral researcher the IMDEA Networks Institute in December 2016 right after having obtained his Ph.D. degree at the University of Luxembourg. He received the Bachelor Degree in Ingegneria Telematica in 2010 and the Master Degree in Computer and Communication Networks Engineering in 2012 both from Politecnico di Torino. Claudio also holds the 2016 SmartICT Certificate on standardization for business innovation from the joint program of University of Luxembourg and ILNAS, the National Standardization Agency. Claudio has been awarded with the Spanish Juan de la Cierva grant and the Best Paper Awards in IEEE Cloudnet 2016 and in ACM WiNTECH 2018 for its work with openLEON. He is member of IEEE and ACM, served as Publication and Web Chair at IEEE CloudNet 2014, Publicity Chair in ACM/IEEE ANCS 2018, Workshop Co-Chair of MoCS 2019 and TPC Co-Chair of IEEE CAMAD 2019. Claudio joined the Editorial Board team of IEEE Networking Letters as of June 2019. His primary research interests include multi-access edge computing, ultra-reliable and low latency communications and mobile crowdsensing.

Guido Cantelmo is a PostDoc Research Associate at the Technical University of Munich (TUM) since January 2019. He holds a diploma in Civil Engineering (2011) and an M.Sc degree in Road Design and Transportation Systems Engineering (2013), both from University of Roma Tre (Rome, Italy). Guido received his PhD from the University of Luxembourg in January 2018. After, he has been working as PostDoc Research Associate at the University of Luxembourg for one year.
In 2014, he has been visiting research student at the KU Leuven (Katholieke Universiteit Leuven). Guido has been awarded a co-funded Eurotech-Marie Skłodowska-Curie individual fellowship to study individual and shared mobility services. His research interests are Mobility as a Service (MaaS), ITS, transportation research, and demand modelling.

Dzmitry Kliazovich is a Head of Innovation at ExaMotive. He was a Senior Scientist at the Faculty of Science, Technology, and Communication of the University of Luxembourg. Dr. Kliazovich holds an award-winning Ph.D. in Information and Telecommunication Technologies from the University of Trento (Italy). His works on cloud computing, energy-efficiency, indoor localization, and mobile networks received IEEE/ACM Best Paper Awards. He coordinated organization and chaired a number of highly ranked international conferences and symposia, including the IEEE International Conference on Cloud Networking (CloudNet 2014). He is the Associate Editor of the IEEE Communications Surveys and Tutorials and of the IEEE Transactions of Cloud Computing journals. His main research activities are in the field of intelligent transportation systems, telecommunications, cloud computing, and Internet of Things (IoT).

SPECIAL ISSUE ON Future Network Architecture, Technologies, and Services

Guest Editor: Mohamed Faten Zhani, ÉTS Montreal, Canada
{mfzhani}@etsmtl.ca

This special issue of Frontiers focuses on Future Network Architecture, Technologies, and Services. This topic is currently gaining momentum and several research teams are working to reinvent the Internet architecture, protocols and technologies in order to adapt it to the requirements of future multimedia applications like virtual and augmented reality, holoportation, and telepresence. For this special issue, we invited leading research groups to submit two papers on the topic in order to highlight their latest achievements and provide their insights on the key research challenges and directions. In the following, we briefly summarize the main contributions of these papers.

The first paper by L. Dong and R. Li entitled "Big Packet Protocol: Advances the Internet with In-Network Services and Functions" presents an overview of the key challenges and requirements of future Internet services and applications. The authors also summarize the recent research of the authors on how to address those requirements, mainly through the Big Packet Protocol, a novel protocol that extends the current IP protocol with new features including the possibility to include commands and Metadata in packets. The article presents two scenarios showing how this new protocol could help to support the requirements of future applications.

In the second paper entitled "Towards 6DoF Virtual Reality Video Streaming: Status and Challenges", J. V. D. Hooft, M. T. Vega, T. Wauters, H. K. Ravuri, C. Timmerer, H. Hellwagner, and F. D. Turck focus on Virtual Reality video streaming as it is one of the most prominent applications of the future. The authors provide an overview of the existing solutions for the delivery of immersive video with six degrees of freedom (6DoF) including image-based and volumetric media-based solutions. The article highlights and discusses key research challenges and opportunities to address 6DoF virtual reality video streaming including content representation and encoding, rate adaptation algorithms, application layer protocols, and in network optimizations.

The purpose of this special issue is to introduce several state-of-the-art research efforts and future challenges pertaining to Future Network Architectures, Technologies, and Services. The valuable contributions of abovementioned renowned researchers make the special issue an excellent reference for the readers. The guest editor is thankful for all the authors for their contributions and their support to the MMTC Communications – Frontiers Board.

Mohamed Faten Zhani is an associate professor with the department of software and IT engineering at l'École de Technologie Supérieure (ÉTS Montreal) in Canada. His research interests include cloud computing, network function virtualization, software-defined networking and resource management in large-scale distributed systems. Faten has co-authored several book chapters and research papers published in renowned conferences and journals including IEEE/IFIP/ACM CNSM, IEEE/IFIP IM/NOMS, IEEE INFOCOM, IEEE transactions on cloud computing and IEEE Journal on Selected Areas in Communications (JSAC). He served as the general or technical program chair of several international workshops and conferences. He is also co-editor of the IEEE Communications Magazine series on "Telecom Software, Network Virtualization, and Software Defined Networks", associate editor of the IEEE transactions on network and service management and of the Wiley international journal of network management, and managing editor of the IEEE softwarization newsletter. He is co-founder and vice-chair of the IEEE Network Intelligence Emerging Technology Initiative and a cluster lead at the IEEE P1916.1 SDN/NFV Performance standard group.

Big Packet Protocol: Advances the Internet with In-Network Services and Functions

Lijun Dong, Richard Li
Futurewei Technologies Inc., Santa Clara, CA, U.S.A
ldong@futurewei.com; richard.li@ futurewei.com

1. Introduction

Most traditional communication media, including radio, television, paper mails and newspapers are reshaped, redefined, or even bypassed by the Internet, giving birth to new medias, which have been globally accessed by us on the everyday basis, such as email, e-banking, online shopping, digital audio/video, social media and IP based phone calls as shown in Figure 1. Internet has tremendously facilitated our daily lives.

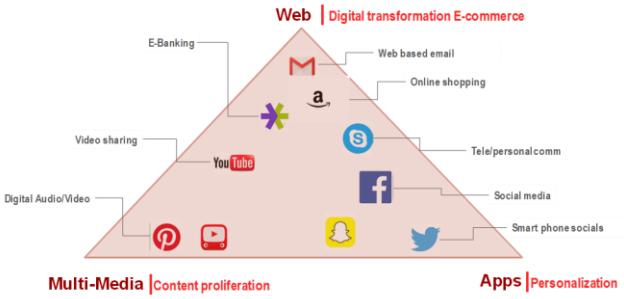


Figure 1. Current Multimedia Content and Internet Applications

What is envisioned to happen to the Internet in the next 10 to 15 years [33]? There will be abundant bandwidth everywhere, everything will be connected to the Internet, new medias and critical applications emerge (Figure 2). Enabled by Internet of Things (IoT), our living environment will become more intelligent and personalized, such as smart city, smart home and smart building. The media will become more enriched and immersive, such as 3D teleconferencing, holographic communication. The applications requiring extreme low latency, such as remote surgery, factory automation and self-driving vehicle will significantly change people's ways of living.

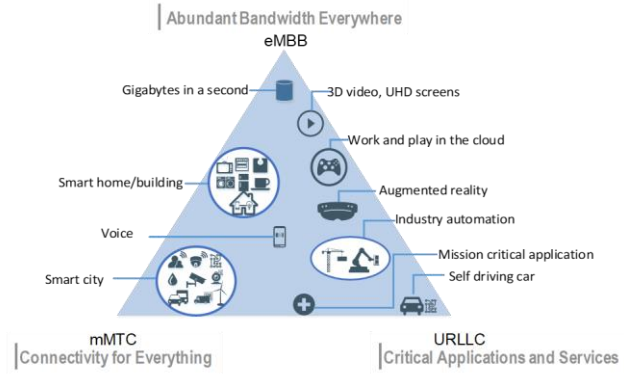


Figure 2. Features of Internet in 2030

The number of IoT devices that connect to the Internet has grown explosively in a global scale in the past years, which will keep growing at a more rapid speed in the future. Ultimately, everything will be connected to the Internet and become accessible anywhere in the world. The trillions of things will generate huge amount of traffic in the Internet. One major requirement is how to reduce the data exchange from and to IoT devices in order to improve the lifetime of the devices and network resource efficiency.

Machine to Machine (M2M) communications have several outstanding properties. Machines never sleep and can be always online if they have enough power. They can generate images and videos with higher resolution, which is beyond what human eyes can perceive. The information processing needs to be extremely fast, the reaction and

feedback time from different entities needs to be synchronized and coordinated. The latency needs to be guaranteed with high precision.

On the factory floor, robots, motion controllers, Programmable Logic Controllers (PLCs), and other manufacturing equipment work together in a synchronized fashion as products roll down the assembly line. This requires making everything to work with strict pre-defined pattern, which is generally dependent on real-time and deterministic performance of the equipment. Real time indicates a system is able to react rapidly enough to service all critical events. Determinism is a measure of the variation in a system's response time to a particular event. One measurement is the jitter performance, which is defined as the worst-case execution time minus the best-case execution time.

Consequently, new network services and functions are needed to enable those applications. For instance, deterministic low latency services means that the latency experienced by the user will be ensured by the network to be within a certain deadline, or at a specific time period. In-network computing will push the computation resources nearer to the end users. The underlying network infrastructure will become more diversified and complementary to each other to provide interconnection among people, devices and any other accessible things.

The current Internet infrastructure provides only the three following services:

- Best effort is the first and most widely used service, in which QoS (Quality of Service) is not provided. Reliability is guaranteed by and only by retransmission. Neither its throughput nor its latency is guaranteed.
- Differentiated Service (DiffServ) has up to 8 classes of services. The services are differentiated hop by hop instead of the whole forwarding path. There is no quantitative control or assurance for its end-to-end behavior. Neither its throughput nor its latency is guaranteed.
- Traffic Engineering (TE) guarantees explicit paths and bandwidth, as well as provides fast-reroute in MPLS (Multiprotocol Label Switching) networks. But neither its throughput nor its latency is guaranteed. There is no differentiation among different types of traffic, nor service quality (e.g. reliability, bandwidth) from IP layer, which needs upper layers to partially resolve service quality issues.

Unfortunately, these services are not sufficient for several future applications, which have new requirements like a zero jitter, time-guarantees, lossless delivery and high throughput. Recently, some technologies have been developed to further improve these basic services but they are still limited. For instance, IPv6 [35] only changes the addressing scheme. Segment Routing Version 6 (SRv6) [36] provides only programmable source routing. MPLS provides only a way to implement traffic engineering and Virtual Private Network (VPN) services. The MPLS RSVP-TE (Resource Reservation Protocol Traffic Engineering) [37][38] guarantees bandwidth, but does guarantee neither throughput nor latency. Software Defined Networking (SDN) [39][40][41] changes the way to control networks. Network Function Virtualization (NFV) [42][43] changes the way to implement network functions. However, none of the above technologies change the nature of the Internet and the traditional statistical multiplexing and best effort forwarding. The underlying IP protocol is still the same for the last 50 years. Today's Internet is therefore not ready for the next decade.

2. Big Packet Protocol (BPP)

IP datagram used to be called as "lettergram" in its early history, and it has many similarities with the postal service. However, today's postal service has evolved. It is now possible to track a package online and know its precise arrival time. It is also possible to customize the way the package is delivered and to ask for a signature at the delivery to ensure the acknowledge it.

With the target of offering similar features, we have proposed an evolutionary framework, called Big Packet Protocol (BPP) [44]. BPP extends the current IP packet and brings minimal changes to the current Internet protocols. The in-network processing of packets is guided by commands and metadata carried in the BPP packet block. The "Commands" could describe how the routers treat the packet on each hop as it traverses the network, e.g., when to drop, how to prioritize, what to collect, how to forward and how to run function on the packet. The "Metadata" contains data about the packet, e.g., the contextual information about the user and the application. The in-network node intelligence is naturally embedded and supported by the BPP framework.

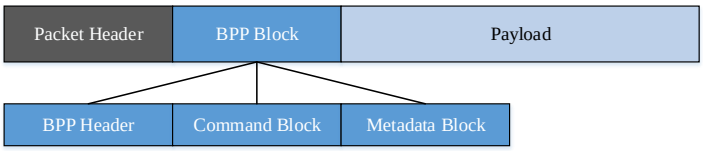


Figure 3. BPP Packet Format

BPP could enable high-precision networking services by adding customized packet processing rules and incorporating dynamic and context-awareness. BPP provides service-level guarantees thanks to in-network measurements and per-hop decision making. BPP commands only affect the behavior of the packets that carry them, not device as a whole or any other packets/flows. Given that the future Internet routers especially the edge routers will be equipped with more storage, computation and processing capabilities, BPP can bring the intelligence of user experience, continuity and awareness of services into the network. BPP block may be added at the source, encapsulated on a BPP boundary (ingress gateway), or inserted in a trusted domain as shown in Figure 4. When BPP header is inserted at the source node, it has to be a BPP aware device (BPP host).

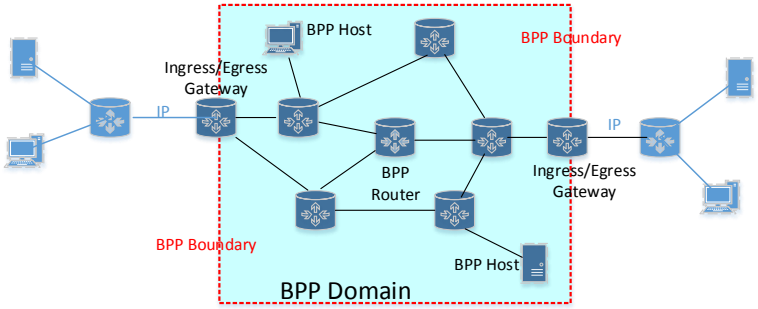


Figure 4. BPP Deployment

3. New Network Functions and Services Facilitated by BPP

In this section, we present two examples showing how BPP facilitates new network functions and services, based on our previous and on-going research work.

3.1 In-Network Semantic Mashup Function

In the domain of web services, mashup refers to combining multiple basic web services in order to create a new one. Similarly, the mashup technique is also used to create a new M2M/IoT service in the M2M/IoT realm.

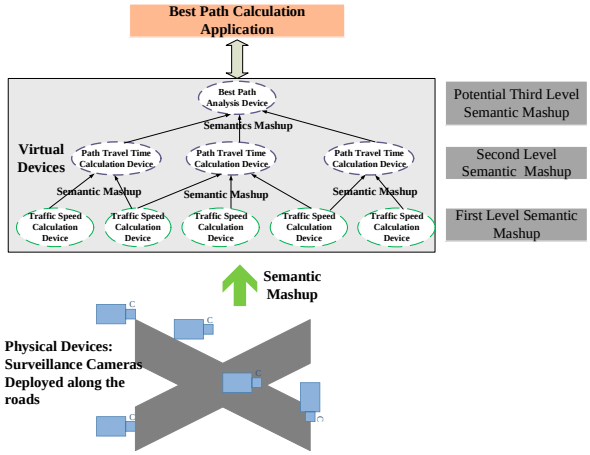


Figure 5. Semantic Mashup Use Case

In the existing semantic of M2M/IoT systems [45], a M2M/IoT application can create and publish "virtual things" that act similar to physical resources and could generate new information. Figure 5 shows an example of IoT video surveillance use case where surveillance cameras deployed along the roads could capture videos for car traffic videos.

Through semantic mashup, the videos could generate the new virtual data like the average traffic speed on the road where the surveillance camera is deployed [46]. Additionally, through the semantic mashup of the traffic speed data, the average travel time from source to destination can be generated as another higher-level virtual data. The semantic mashup can also be applied to compare the average travel time of alternative paths for the same source and destination pair, which gives the best path with the least travel time.

Although semantic mashup has been provided as a service offered by servers deployed in the existing M2M/IoT systems [47], this architecture has several limitations: (1) the semantic mashup is done in a centralized way by each by the the server, which means the original raw data provided by the physical devices need to be retrieved by these servers or from each device, (2) it does not consider there could be cached copies of original data in the network, thus rules out the option that a router may be able to route the requests to nearer cached copies. (3) it does not consider that a router may carry out the in-network semantic mashup without transporting the original big-sized data to the Common Service Server. Those limitations result in an increase of the user's experienced latency and in a wastage of the network bandwidth.

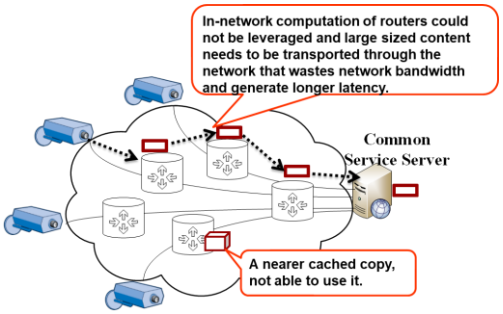


Figure 6. Inefficiencies of Overlay based Semantic Mashup

BPP could enable opportunistic semantic mashup in the routers by leveraging a BPP packet as shown in Figure 7 [48][49][50]. The BPP Command is set up as "SemanticMashup". The Metadata block includes the parameters that are needed for in-network semantic mashup function. The first field is the ultimate target data name (N^{th} level data) if it is known to the requester, which may not be available in the network and requires semantic mashup from other data. The second field can optionally include the semantic mashup logic for the Nth level data. The next field is to specify the $(N\text{-}1)^{th}$ level data that is used for semantic mashup to generate the Nth level data. Similarly, the $(N\text{-}1)^{th}$ level data may also be generated from the semantic mashup on other data, whose logic can be included in the following field. This can continue till the level 0, which is the raw data produced by the IoT devices. Ideally, in-network semantic mashup should happen for 1 or 2 levels at most, considering the overhead caused by the request message transmission, as well as the processing overhead and latency introduced at the forwarding routers.

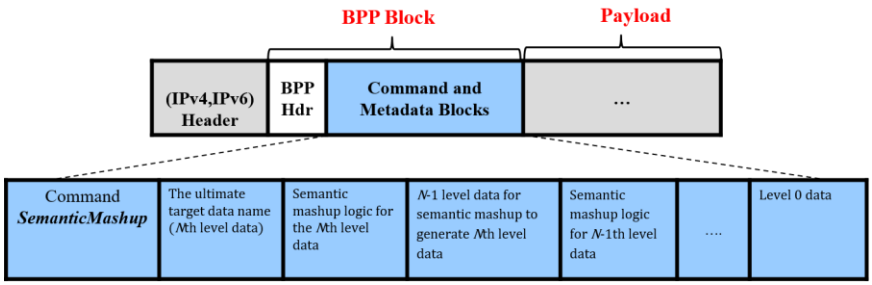


Figure 7. BPP Packet Facilitating In-Network Semantic Mashup Function

The Mashed-up result can be returned by one of the intermediate routers, which significantly reduces the data size, as well as the number of hops that the original raw data traverses in the network. It enables offloading the computation overhead from the middleware server or the user terminals, also alleviating the transmission overhead from the network. As a result, the latency in getting the target data significantly decreases and the improvement increases with the number of data involved in semantic mashup.

3.3 Qualitative Communication Service

In the current Internet, a packet is a minimal, self-contained unit of delivery that gets transmitted, classified, or

discarded without any discrimination. In other words, a packet is treated as an atomic unit over which network actions are performed. As a result, the integrity of the information contained in the packet payload is verified at the packet level as well. Checksum computation is carried out on the entire packet.

A packet could be dropped entirely when the congestion occurs and the buffer space becomes full, and also when a part of the packet is corrupted due to lossy links. When TCP is used, packet loss results in the retransmission of the packet. The retransmission of packets incurs unpredictable latency, as well as an increase in the network load, consuming extensively more network resources. The retransmitted packet has to re-travel the path towards its destination. On the other hand, the sender is not aware that the packet has been dropped until a timeout or duplicate acknowledgments are received. This delay extends the time at the sender side before the retransmission can be initiated, leading to additional latency and a reduced network throughput.

Currently, bits in the stream have no meanings, the routers treat them indiscriminately. However, applications perceive information semantically, which means that the semantics information associated with the data payload could imply what portions are inter-related or are more significant. BPP can make different pieces of the packet become independent logic units, which are called chunks. By including the semantics associated with different chunks in the packet, the network could perform finer granularity of operations on the packet. One exceptional benefit of the chunk concept is that selective chunk dropping is allowed in the network. When network conditions arise that would currently require the whole packet dropping, it is allowed for the network to drop those insignificant or less prioritized chunks instead of the whole packet. The packet payload can be preserved as much as possible for those more significant chunks.

Qualitative Communication Service [51] is defined as a packetization scheme that breaks down the payload into multiple chunks, each with certain semantics or significance. The routers make decisions to selectively drop chunks based on the current situation and the significance carried in the packet. A packet under the qualitative communication service is called a qualitative packet.

With Qualitative Communication Service, a packet retransmission may not be required if the receiver has the capability to recover the original data or is satisfied with the information left in the packet after removal of certain chunks from the payload by the intermediate routers. In this case, the receiver can acknowledge the acceptance of the packet, while it may also indicate to the sender that it was partially dropped in the network. Network resource usage can be reduced and better prioritized for the delivery of other packets. The throughput for an individual data flow is the amount of data moved successfully from one place to another in a given time period. The qualitative communication service allows the network to deliver more important information in the packets to the destinations, by preventing the packets from being discarded completely. Therefore, the throughput is less jeopardized by packet loss, and is capable of reaching a higher effective throughput rate.

The proposed Qualitative Communication Service requires the support from the application layer, transport layer and network layer. Only an application can tell what data can be treated qualitatively and how to treat it. Thus, applications need to feed metadata to the network layer. The partial dropping of a packet should be informed to and understood by the sender as a warning that some level of congestion is occurring in the network. The network layer needs a well-formed metadata to conditionally perform operations in a hop-by-hop manner.

In Qualitative Communications, Packet Wash refers to a scrubbing operation that reduces the size of a packet while retaining as much information as possible. BPP could facilitate the proposed Packet Wash operation by setting it up in the Command block, with the necessary parameters included in the Metadata block. Different chunks in a packet payload can either have different priority/significance or equally important. For instance, the order of the chunks in the qualitative packet could be re-arranged such that the least significant chunks are always in the tail, which makes the Packet Wash operation easier to be performed (i.e. always drop the chunks in the tail first). In this case, the original order of the corresponding chunk should be carried in the packet and informed to the receiver. The BPP Metadata block needs to include the following parameters as shown in Figure 9:

- Condition: under which the Packet Wash operation is triggered, for example the router's outgoing buffer is 90% full.
- Threshold: beyond which the chunks in a packet could not be dropped anymore, otherwise the packet would be considered useless according to the source of the packet.
- Checksum CRC_i : it is used to verify the integrity of the chunk *i*. In other word, each chunk has its own checksum.

- Relative offset Off_i : it describes the boundary between two adjacent chunks.
- Original order ORD_i : it indicates the original position of the chunk *i* in the data payload.
- Flag OF_i: it indicates whether the chunk *i* was dropped or not during the transmission.

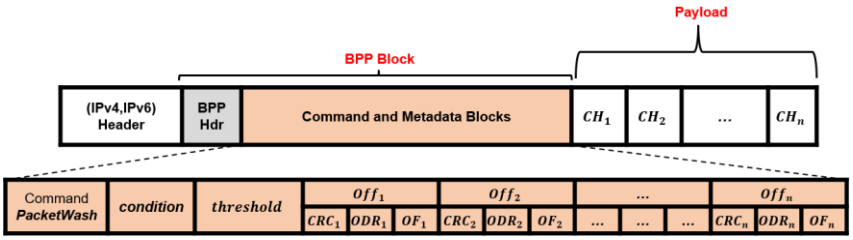


Figure 9. BPP Packet for In-Network Packet Wash for Chunks with Different Significance

Another example is that of a random linear network coding that is applied to the chunks in the packet payload [52]. The network-coding granularity is applied to the chunks instead of the packets. Since the linear-network-coded chunks can equally contribute to the original packet payload decoding, they have the same importance/significance. The sender will include the same number of independent coded chunks as the packet payload. BPP could be leveraged to carry the necessary information for the receiver to decode the packet payload, which may go through the Packet Wash operation in the network. The BPP Metadata block needs to include the following parameters as shown in Figure 10:

- PID: it is used to identify the packet identifier for a specific data content originated from a sender. PID only relates to the data content and the sender. PID is used to provide a way to match a cached chunk of a data content.
- flag: it indicates whether the payload contains linear-network-coded chunks or not.
- size: it shows how large the chunk size is. When partial packet dropping happens, the network nodes are able to find the boundary of each coded chunk in the payload.
- coefficients: they are the coefficients with which source chunks are linearly combined to form coded chunks contained in the current payload.
- currentRank: it indicates the current rank of the remaining coded chunks in the payload.
- fullRank: it indicates the full rank of the coded chunks in the payload when they are inserted by the sender.

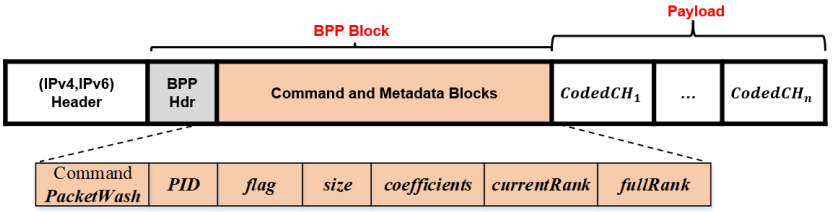


Figure 10. BPP Packet for In-Network Packet Wash for Linear Network Coded Chunks

5. Conclusion

This paper presented our understanding on the challenges and requirements of the future Internet services to satisfy the new applications in the next decade. It gives a short summary of our recent researches on how to address those requirements, mainly we proposed the Big Packet Protocol that extends the current IP protocol with a BPP block, which includes "Command" and "Metadata" fields. A BPP enabled network node could act on those commands and metadata to handle the packet, overriding any "regular" packet processing logic. By leveraging BPP, future Internet is more intelligent and application oriented to satisfy stringent requirements like deterministic latency guarantee.

This letter also describes two examples that illustrate the benefits of BPP. The first example shows that semantic mashup function can be carried out in the Internet routers, which could more likely be the edge routers to offload the computation overhead from the middleware server or the user terminals, to alleviate the transmission overhead from the network, as well as to reduce the user's perceived latency. The second example is a new type of packetization

method called Qualitative Communication Service, through which finer granularity of packet processing is allowed in the network, yielding less packet loss ratio and retransmissions, higher throughput, and less latency.

References

[33] "IMT Vision – Framework and Overall Objectives of the Future Development of IMT for 2020 and Beyond," https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.2083-0-201509-I!!PDF-E.pdf, Sept. 2015.

[34] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, "RFC 2475: An Architecture for Differentiated Services," IETF, Dec. 1998.

[35] S. Deering, R. Hinden, "RFC 8200: Internet Protocol, Version 6 (IPv6) Specification," IETF, Jul. 2017.

[36] Segment Routing IETF Proceedings, https://www.segment-routing.net/ietf/.

[37] D.O. Awduche, "MPLS and Traffic Engineering in IP Networks," IEEE Communications Magazine, vol. 37, no. 12, pp. 42–47, Dec 1999.

[38] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels," IETF RFC 3209.

[39] B. N. Astuto, M. Mendonça, X. N. Nguyen, K. Obraczka, T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks," IEEE Communications Surveys and Tutorials, vol. 16, no. 3, pp. 1617–1634, 2014.

[40] K.Benzekki, A. E. Fergougui , A. E. Elalaoui, "Software-Defined networking (SDN): A survey". Security and Communication Networks. vol. 9, no. 18, pp. 5803–5833, Dec. 2016.

[41] W. Xia, Y. Wen, C. H. Foh, D. Niyato, H. Xie, "A Survey on Software-Defined Networking," IEEE Communication Surveys and Tutorials, vol. 17, no. 1, pp. 27-51, First Quarter 2015.

[42] Y. Li, M. Chen, "Software-Defined Network Function Virtualization: A Survey," IEEE Access, no. 3, pp. 2542–2553, Dec. 2015.

[43] B. Yi, X. Wang, K. Li, S. K. Das, M. Huang, "A Comprehensive Survey of Network Function Virtualization," Computer Networks, no. 133, pp. 212–262, 2018.

[44] R. Li, A. Clemm, U. Chunduri, L. Dong, K. Makhijani, "A New Framework and Protocol for Future Networking Applications," ACM Sigcomm NEAT Workshop, 2018.

[45] L. Dong, D. N. Seed, G. Lu, "Semantics Support and Management in M2M Systems," US20140330929A1.

[46] F. Mehboob , M. Abbas , R. Jiang, "Traffic Event Detection from Road Surveillance Videos Based on Fuzzy Logic," 2016 SAI Computing Conference.

[47] oneM2MFunctional Architecture TS 0001, http://www.onem2m.org/technical/published-drafts.

[48] L. Dong, R. Li, "Offloading Semantic Mashup by On-Path Routers for IoT Applications," IEEE PIMRC 2018.

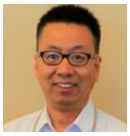
[49] L. Dong, R. Li, "Support Opportunistic En-Route Information Mashup of IoT Data," IEEE SmartIoT 2018.

[50] L. Dong, R. Li, "Enhance Information Derivation by In-Network Semantic Mashup for IoT Applications," IEEE EuCNC 2018.

[51] R. Li, K. Makhijani, H. Yousefi, C. Westphal, L. Dong, T. Wauters, F. D. Turck, "A Framework for Qualitative Communications Using Big Packet Protocol," ACM Sigcomm NEAT Workshop, 2019.

[52] L. Dong, R. Li, "In-Packet Network Coding for Effective Packet Wash and Packet Enrichment," IEEE Globecom Workshop FI2030, 2019.

Lijun Dong received her Ph.D. degree in Electrical and Computer Engineering and M.S degree in Statistics from Rutgers University. She is a Senior Staff Researcher at Futurewei Technologies Inc. USA. She has broad and in-depth researches in the areas of Internet of Things, Machine-to-Machine communications, Information-Centric Networking and Future Internet Architecture for more than a decade. She has served for many international conferences: publicity chair of CSCN 2016, TPC co-chair of ICNC 2017, Industry Program Chair of ICNC 2020, TPC member of Globecom 2011, ICNC 2012 and 2013, iThings 2013, ISCC 2017 and 2018, HiPNet 2018 and 2019. She is one of the board members of the WOCC conference. She has been an active and influential contributor, and responsible for internal strategy development and execution for standards, including oneM2M, IETF, 3GPP and ITU. She is the major inventor to 36 granted patents and 56 pre-granted patents. She has 40+ journal and conference publications.



Richard Li is Chief Scientist and Head of Network Technologies Lab at Futurewei Technologies Inc. USA, where he leads a senior research team to design and develop next-generation network architectures, technologies, protocols, and solutions. Meanwhile, Richard serves as the Chairman of the ITU-T FG Network 2030, the Vice Chairman of the European ETSI ISG NGP (Next-Generation Protocols), Co-Chairs of Technical Program Committees for some ACM/IEEE/IARIA conferences and workshops. Prior to joining Futurewei, he worked with Cisco and Ericsson in his various capacities in the field of networking technologies, standards, solutions and operating systems. During his career, Richard spearheaded network technology innovation and development encompassing several areas of networking such as Routing and MPLS, Mobile Backhaul, Metro and Core Networks, Data Center, Cloud and Virtualization. Richard is extremely passionate about advances in data communications, and challenges himself by solving problems in their entirety thus creating a bigger and long-term impact on the networking industry.

Towards 6DoF Virtual Reality Video Streaming: Status and Challenges

Jeroen van der Hooft[1], *Maria Torres Vega*[1], *Tim Wauters*[1], *Hemanth Kumar Ravuri*[1], *Christian Timmerer*[2], *Hermann Hellwagner*[2], *Filip De Turck*[1]
[1]*IDLab, Department of Information Technology, Ghent University - imec*
[2]*MMC, Institute of Information Technology, Alpen-Adria-Universität Klagenfurt*
jeroen.vanderhooft@ugent.be

1. Introduction

In the last few years, delivery of immersive video with six degrees of freedom (6DoF) has become an important topic for content providers. Recent technological advancements have resulted in affordable head-mounted displays, allowing a broad range of users to enjoy Virtual Reality (VR) content. Service providers such as Facebook[1] and YouTube[2] were among the first to provide 360° video, using the principle of HTTP Adaptive Streaming (HAS) to deliver the content to the end user. In HAS, the content is encoded using several quality representations, temporally segmented into chunks of one to ten seconds and stored on one or multiple servers within a content delivery network. Based on the perceived network conditions, the device characteristics, and the user's preferences, the client can then decide on the quality of each of these segments [1]. Having the ability to adapt the video quality, this approach actively avoids buffer starvation, and therefore results in smoother playback of the requested content and a higher Quality of Experience (QoE) for the end user [2]. The introduction of 360° video provides the user with three degrees of freedom to move within an immersive world, allowing changes in the yaw, roll, and pitch. In the last few years, multiple solutions have been proposed to efficiently deliver VR content through HAS, focusing, for instance, on foveas- and tile-based encoding, improved viewport prediction (i.e., prediction of the user's head movement in the near future in order to buffer useful high-quality content), and application layer optimizations [3]. In these works, however, the location of the user remains fixed to the position of the camera within the scene. Recently, significant research efforts have been made to realize 6DoF for streamed video content, i.e., the user may experience three additional degrees of freedom by being able to change the viewing position in a video scene. These efforts are promising, but significant research contributions will be required in order to realize its full potential. In this paper, an overview of existing 6DoF solutions is presented, and key challenges and opportunities are highlighted.

2. 6DoF Video Solutions

Two types of approaches to implement 6DoF video solutions are generally considered: image-based and volumetric media-based solutions. The former requires a representation of images at every different angle and tilt, while the latter stores objects as a collection of points in the three-dimensional space.

2.1. Image-Based Solutions

In image-based solutions, the system renders different views of an environment from a set of pre-acquired imagery. Images are typically captured using camera arrays (see Figure 7) or cylindrical camera setups, resulting in a representation of images at every different angle and tilt. Because different representations are immediately available, displaying content corresponding to a given position and viewing direction requires modest computational resources [4]. However, the approach results in large storage and bandwidth requirements, since roughly every 0.3-degree difference in angle requires a new image in order to provide a smooth transition between images [5].

While the concept of light fields has been around for two decades, it recently caught more attention as a means to provide 6DoF content streaming; in the last few years, several image-based solutions and frameworks have been proposed. Wijnants et al. propose a DASH-compliant framework for the delivery of light fields [6]. The authors achieve real-time rendering by leveraging video decoding to contemporary consumer-grade GPUs, using disk-versus-GPU caching in order to render source images more quickly. While the proposed framework allows the user to move around and download content in an adaptive manner, only static light fields are considered; delivery of video content is not supported. In addition, only single objects are studied in their work. Daniel et al. propose SMFoLD, an open

[1] https://www.facebook.com, last accessed: August 19, 2019
[2] https://www.youtube.com, last accessed: August 19, 2019

streaming media standard for light field video [7]. This standard allows compliant displays to receive a stream of three-dimensional frame descriptions and render scenes without the need for specialized head-mounted devices. The authors discuss several technical challenges to realize a fully functional system framework, focusing on encoding, streaming and displaying the three-dimensional content. Kara et al. propose a framework for subjective evaluation of light field scenes, considering different spatial resolutions and angular differences between images [8]. Similar to traditional video, the authors show that quality switching is preferred over long stalling events. Furthermore, they conclude that it is beneficial to choose a video representation with both spatial and angular resolution reduced, if the smoothness of the continuous horizontal motion parallax cannot be guaranteed. While these research efforts offer interesting insights into the possibilities of light fields for 6DoF content streaming, an operational framework for high-quality image-based 6DoF video streaming does not yet exist.

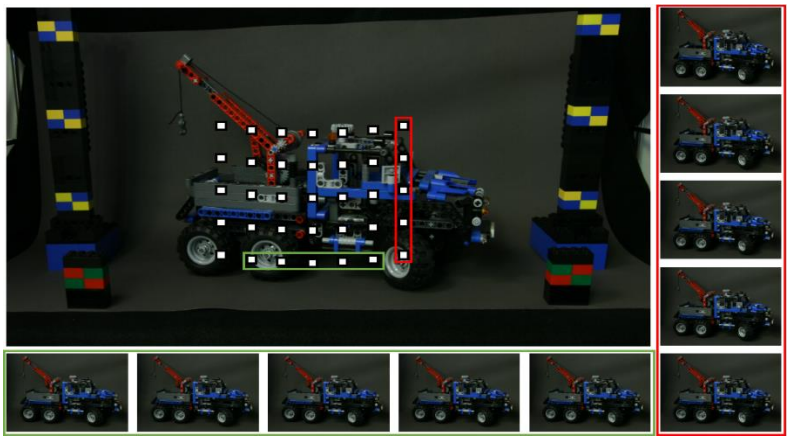


Figure 7. Illustration of light field capture from the (New) Stanford Light Field Archive [9].

2.2. Volumetric Media-Based Solutions

Volumetric media-based solutions store objects as a collection of points. Capturing the geometry (*x*, *y*, *z* tuples) and color (*RGB* values) of thousands or millions of points, the object can be rendered from any viewing angle [10]. This reduces storage and bandwidth costs, but requires complex preprocessing (i.e., multiple camera angles and depths) and rendering at client-side. An example of a capture setup and a generated point cloud object are shown in Figure 8.

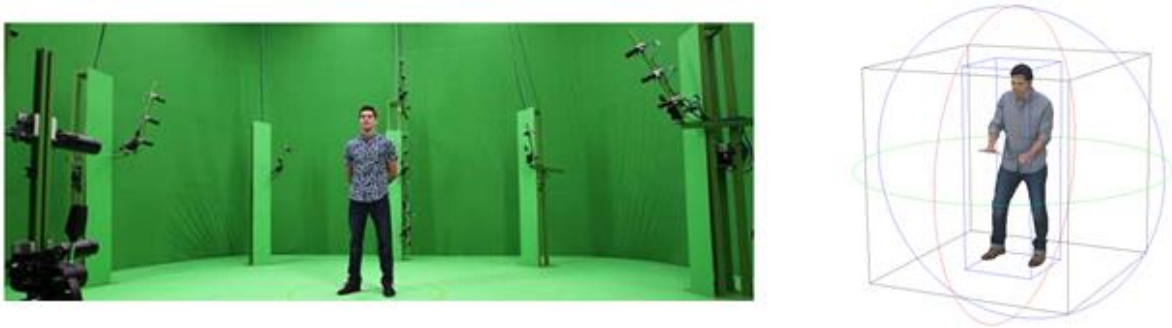


Figure 8. Example point cloud capture from the reference MPEG dataset [11].

Focusing on video on demand, Hosseini and Timmerer are the first to propose a DASH-compliant approach for single point cloud streaming [12]. Rather than using a dedicated encoder, the authors sample the different points to generate versions of lower quality. Furthermore, objects are requested on a per-frame basis, which means that a number of HTTP GET requests proportional to the frame rate is required. He et al. consider view-dependent streaming of point cloud objects, using cubic projection to create six two-dimensional images which can then be compressed using traditional compression techniques [13]. The proposed approach relies on a hybrid network (broadband and broadcast) and in-network optimizations such as caching. In a previous work [14], we proposed PCC-DASH, a framework for streaming 6DoF scenes consisting of multiple point cloud objects. Point cloud compression is used to prepare multiple quality versions of the considered objects, and several rate adaptation heuristics are proposed which take into account the user's position and viewing angle. However, the framework is not able to decode and render the content in real-

time.

Figure 9. Experimental setup by Dijkstra-Soudarissanane et al. [15]. Two users are immersed in a virtual world, in which they can both see and hear one another in real-time.

Concerning live streaming, Dijkstra-Soudarissanane et al. propose a multi-view end-to-end system for real-time capture, transmission and rendering of volumetric media [15]. This system relies on a multi-point control unit (MCU), to shift processing from end devices into a server. Through a relevant demonstrator, the authors show that two end users can effectively communicate within an immersive world (see Figure 9). Similarly, Qian et al. propose Nebula, a volumetric video system that leverages edge computing to reduce computational efforts on the user's device [16]. The setup uses regular pixel-based video encoding and decoding, in order to stream and render a single point cloud object at the client side. Both solutions show that offloading the decoding to the edge results in timely decoding of smaller point cloud objects, which is of major importance for future volumetric media applications.

3. Challenges and Opportunities

Early 6DoF solutions have shown promising results, but are limited in terms of interactivity and complexity. Below, we discuss different challenges that need to be addressed to fulfill the true potential of 6DoF video streaming. As shown in Figure 10, these challenges focus on i) content representation and encoding, ii) rate adaptation algorithms, iii) application layer protocols, iv) in-network optimizations and v) enhanced evaluation techniques.

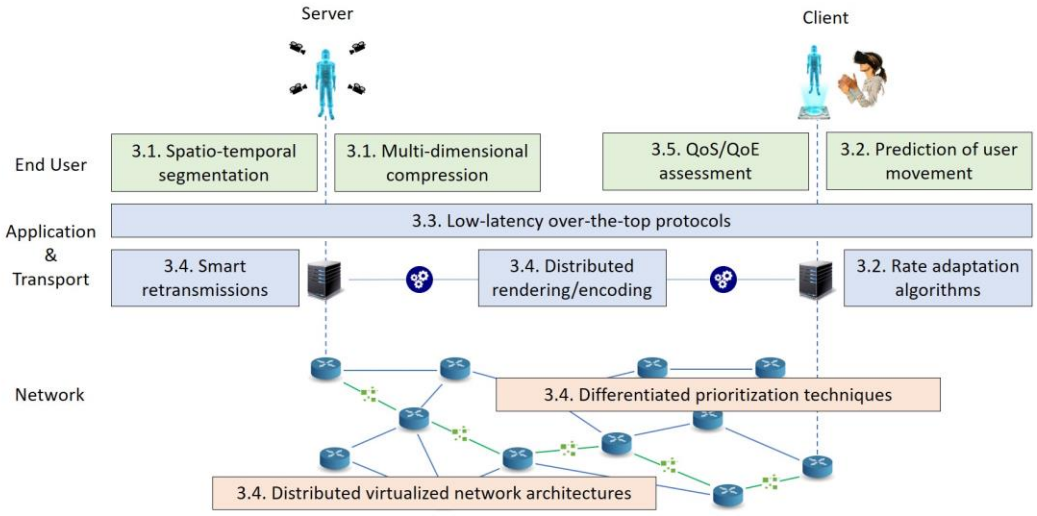


Figure 10. Challenges faced to enable 6DoF video streaming (adapted from [5]).

3.1. Content Representation and Encoding

A key aspect in enabling 6DoF video streaming is the representation of the considered content. Since 2016, the MPEG project on "Coded representation of immersive media", or MPEG-I, has taken on the challenge of enabling coding, transmission and presentation of immersive media [17]. Efforts have already resulted in the Omnidirectional MediA Format (OMAF), which is currently being extended to support interactivity and 3DoF+, to enable limited

modifications of the viewing position. For 6DoF solutions, however, no new formats have been proposed at the time of writing. Therefore, traditional image- and volumetric media-based solutions are still widely adopted.

As mentioned earlier, image-based solutions require a representation of images at every different angle and tilt. Feasible compression rates can be obtained through spatial reduction (i.e., reducing the resolution of each image) and angular reduction (i.e., reducing the number of images, mostly affecting users with high movement) [8]. Meanwhile, compression techniques for point clouds mostly include static kd-tree- and octree-based solutions, with notable examples including Google's Draco [18] and the work by Schnabel and Klein [19]. However, since MPEG launched its call for proposals in 2017 [20], alternative approaches have been suggested. Following an extensive evaluation of nine submitted proposals, MPEG selected a reference encoder for video-based point cloud compression (V-PCC) [10]. This encoder converts point clouds into two separate video sequences, which capture the geometry and texture information, and applies traditional video coding techniques to compress the data. However, this compression technique cannot be used to decode point cloud objects in real-time on commodity hardware [14].

Currently, most encoding techniques are based on compression standards for traditional images and video. Future research efforts should focus on media formats adapted to the considered use case, for instance through the application of sparse representations [21]. Furthermore, as illustrated by the work by Dijkstra-Soudarissanane et al. [15], offloading the decoding to edge nodes offers the advantage of timely decoding and sharing of resources. Ongoing efforts are needed to optimize this process, possibly combining it with in-network optimizations.

3.2. Rate Adaptation Algorithms

Similar to traditional video streaming, rate adaptation is an important factor when enabling 6DoF video streaming over the best-effort Internet. Its complexity, however, is significantly higher. Rate adaptation algorithms need to take into account both content characteristics and network throughput and latency, as well as the user's movement and viewing angle. In this regard, culling the considered objects (i.e., reducing the amount of content based on what regions of the video the user can observe [22]) is a well-known method to reduce bandwidth requirements.

Recently, a number of rate adaptation heuristics have been proposed. Qian et al. present two rate adaptation mechanisms for Nebula [16], while Hosseini presents a rate adaptation heuristic for multiple point cloud objects [23]. However, they did not carry out an evaluation and no results on the video quality are reported. Park et al. propose a utility-based rate adaptation heuristic for volumetric media, which is both throughput- and buffer-aware [24]. The heuristic was evaluated using simulation, reporting considered utility metric values of each object rather than the resulting visual quality. In our previous work, we propose several rate adaptation heuristics for multi-object point cloud scenes, which consider the user's position and viewing angle within the scene [14]. Results are encouraging, but indicate that the performance of the heuristics strongly depends on the considered video and camera path.

The above heuristics are typically evaluated using fixed user trajectories. This is an idealized scenario, since the client can adapt the quality of the content based on perfect knowledge. Ongoing research will rather have to focus on predicting the user's position and viewing direction, based on the user's (recent) history, video saliency and content. Similar techniques for 360° video already exist [3], but have to be improved to deal with additional complexity.

3.3. Application Layer Optimizations

Nowadays, most HAS solutions use HTTP/1.1 over TCP to retrieve the required resources through request-response transactions, buffering fetched video segments and playing them out in linear order. One possibility to speed up TCP-based solutions is to develop smarter retransmission schemes. In 6DoF video streaming, retransmission is required when transmitting or updating the manifest file or other data crucial for rendering virtual views. However, when less important data is sent (e.g., incremental data or less important frames), the loss may be acceptable. The transport layer needs to handle these different scenarios and decide what to do for each packet, stream or flow.

Some solutions revert to UDP, in order to improve latency at the cost of reliability. One example is the HTTP/3 protocol, which will soon be standardized by the Internet Engineering Task Force (IETF) [25]. This protocol is based on the QUIC protocol, proposed by Google in 2012 [26]. HTTP/3 establishes a number of multiplexed UDP connections, resulting in independent delivery of multiple streams of data. In contrast to HTTP/2, which uses a single TCP connection, this approach avoids head-of-line-blocking if any of the TCP packets are delayed or lost. Another example is WebRTC, a real-time communication protocol which has shown positive results for traditional video in the recent past [27]. WebRTC is, however, peer-to-peer in nature and thus requires multiple encoders at different

qualities for each peering connection to ensure an adaptive streaming solution, which hampers scalability. Existing research on dynamically recomputing encoding settings is very limited and immersive scenarios (with three, let alone six degrees of freedom) have not yet been studied at all.

3.4. In-Network Optimizations

Although various optimizations on higher layers provide support in managing high-bandwidth and low-latency requirements, networks still have a substantial role to play in the end-to-end streaming of 6DoF content. With the advent of multi-camera systems, the computational complexity of tasks such as encoding and rendering increased exponentially, making it difficult to implement them on the end-user equipment. Such tasks can be migrated to resourceful cloud/fog servers on the network. For this reason, networks should be more than just transport circuits.

To support efficient delivery of immersive media services, networks require more programmability. This can be achieved by three evolving technologies: i) Software-Defined Networks (SDN), ii) Network Function Virtualization (NFV) and iii) Multi-access Edge Computing (MEC). The SDN paradigm offers flexibility to the networks in the form of programmable network management, easy reconfiguration and on-demand resource allocation. It suffers, however, from issues such as scalability, control plane overhead and Denial of Service (DoS) attacks. Such shortcomings are to be addressed in order to support 6DoF content streaming. NFV allows the network functions to be deployed as virtualized software entities running on commodity hardware [28]. Various services involved in 6DoF video streaming can be mapped to respective network functions and can be deployed as a service function chain (SFC). The SFC can be distributed to different locations in accordance to various requirements such as hardware capacity, bandwidth, distance, latency, reliability and their respective tradeoffs [28] [29]. As an example, tasks such as view-synthesis can be offloaded from the end-user equipment but should be placed closer to the user in order to lower the latency. This can be achieved by MEC, which enables the devices to access cloud/fog resources in an on-demand fashion [29]. Since 6DoF video streaming depends on diverse factors, the function placement should be treated as multi-objective optimization problem that has not been extensively researched yet. In addition, there is a need for novel and proactive resource management mechanisms for better resource utilization.

The success of content delivery networks increased the prominence of strategic content caching at the edge. Such storage will play an important role in 6DoF content streaming; upon a new task request the server/network needs to swiftly decide if it should store the content for future requests or not. Furthermore, cache placement and distribution is an important research direction. Proactive caching strategies need to evolve, however, as they depend on spatio-temporal traffic predictions, the users' location, mobility, etc. Other network level approaches such as network coding [30] and network slicing [28] can be exploited to meet the requirements of 6DoF video streaming.

3.5. Evaluation Metrics

The perceived quality of 6DoF video depends on many parameters, such as the frame rate and the degree resolution. Therefore, understanding the effects and "sweet spots" of each of the parameters on the user perception is fundamental to help improve the bandwidth consumption while maintaining the user's QoE [5]. Given the subjective essence of the user's experience, in the last years several works have appeared that subjectively assess the QoE of holographic media. Such are the cases of Kara et al. for adaptive streaming of light field video [8] and of Javaheri et al. [31] for point cloud streaming. However, coping with the dynamics of 6DoF video streaming will require real-time measurements of how the user perceives the streaming. In such cases, objective metrics are better suited for the assessment [31]. For this reason, current effort goes in the direction of devising objective metrics for holographic media correlating with the user's QoE. One first step has been to adapt objective metrics traditionally used for two-dimensional videos for virtual reality and 6DoF video content. Variants derived from the mean square error (MSE) and the peak signal-to-noise ratio (PSNR) have shown good correlation both for point cloud [31] and light field video [32]. Despite the promising results, these studies have focused mainly on encoding derived artifacts. Thus, their accuracy to assess the effects of end-to-end system (capturing, encoding and streaming) degradation is still unknown. Furthermore, most of the current studies (both objective and subjective) assume the user to be passive, thus the effects of the interactivity with the holographic content are not taken into account. This will be a fundamental subject of research for future 6DoF applications.

4. Conclusions

This letter presented a brief overview of ongoing research efforts to realize virtual reality video streaming with six

degrees of freedom. Both image- and volumetric media-based representation techniques were discussed, and a list of challenges and opportunities for future work was presented. Given the many applications of virtual reality and an increased interest from both service and content providers, the topic is expected to remain an open field for research and innovation for many years to come.

Acknowledgements

Maria Torres Vega is funded by the Research Foundation - Flanders.

References

[1] A. Bentaleb, B. Taani, A. C. Begen, C. Timmerer and R. Zimmermann, "A Survey on Bitrate Adaptation Schemes for Streaming Media Over HTTP," *IEEE Communications Surveys Tutorials,* vol. 21, no. 1, 2019.

[2] M. Seufert, S. Egger, M. Slanina, T. Zinner, T. Hossfeld and P. Tran-Gia, "A Survey on Quality of Experience of HTTP Adaptive Streaming," *IEEE Communications Surveys Tutorials,* vol. 17, no. 1, 2015.

[3] J. van der Hooft, M. Torres Vega, S. Petrangeli, T. Wauters and F. De Turck, "Tile-Based Adaptive Streaming for Virtual Reality Video," *Submitted to ACM Transactions on Multimedia Computing, Communications, and Applications,* 2019.

[4] M. Levoy and P. Hanrahan, "Light Field Rendering," in *Annual Conference on Computer Graphics and Interactive Techniques*, 1996.

[5] A. Clemm, M. Torres Vega, H. K. Ravuri, T. Wauters and F. De Turck, "Towards Truly Immersive Holographic-Type Communication: Challenges and Solutions," *Submitted to IEEE Communications Magazine,* 2019.

[6] M. Wijnants, H. Lievens, N. Michiels, J. Put, P. Quax and W. Lamotte, "Standards-Compliant HTTP Adaptive Streaming of Static Light Fields," in *ACM Symposium on Virtual Reality Software and Technology*, 2018.

[7] J. R. Daniel, B. Hernàndez, C. E. Thomas, S. L. Kelley, P. G. Jones and C. Chinnock, "Initial Work on Development of an Open Streaming Media Standard for Field of Light Displays (SMFoLD)," *Electronic Imaging,* vol. 2018, no. 4, 2018.

[8] P. A. Kara, A. Cserkaszky, M. G. Artini, A. Barsi, L. Bokor and T. Balogh, "Evaluation of the Concept of Dynamic Adaptive Streaming of Light Field Video," *IEEE Transactions on Broadcasting,* vol. 64, no. 2, 2018.

[9] Computer Graphics Laboratory, Stanford University, "The (New) Stanford Light Field Archive," 2008.

[10] S. Schwarz, M. Preda, V. Baroncini, M. Budagavi, P. Cesar, A. Chou, R. A. Cohen, M. Krivokuca, S. Lasserre, Z. Li, J. Llach, K. Mammou, R. Mekuria, O. Nakagami, E. Siahaan, A. Tabatabai, A. M. Tourapis and V. Zakharchenko, "Emerging MPEG Standards for Point Cloud Compression," *Journal on Emerging and Selected Topics in Circuits and Systems,* vol. 9, no. 1, 2018.

[11] E. d'Eon, T. Myers, B. Harrison and P. A. Chou, "ISO/IEC JTC1/SC29 Joint WG11/WG1 (MPEG/JPEG) input document WG1M40059/WG1M74006. 8i Voxelized Full Bodies - A Voxelized Point Cloud Dataset," 2017.

[12] M. Hosseini and C. Timmerer, "Dynamic Adaptive Point Cloud Streaming," in *Packet Video Workshop*, 2018.

[13] L. He, W. Zhu, K. Zhang and Y. Xu, "View-Dependent Streaming of Dynamic Point Cloud over Hybrid Networks," in *Advances in Multimedia Information Processing*, 2018.

[14] J. van der Hooft, T. Wauters, F. De Turck, C. Timmerer and H. Hellwagner, "Towards 6DoF HTTP Adaptive Streaming Through Point Cloud Compression," in *ACM Multimedia Conference*, 2019.

[15] S. Dijkstra-Soudarissanane, K. El Assal, S. Gunkel, F. ter Haar, R. Hindriks, J. W. Kleinrouweler and O. Niamut, "Multi-Sensor Capture and Network Processing for Virtual Reality Conferencing," in *ACM Multimedia Systems Conference*, 2019.

[16] F. Qian, B. Han, J. Pair and V. Gopalakrishnan, "Toward Practical Volumetric Video Streaming on Commodity Smartphones," in *International Workshop on Mobile Computing Systems and Applications*, 2019.

[17] M. Wien, J. M. Boyce, T. Stockhammer and W. Peng, "Standardization Status of Immersive Video Coding," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems,* vol. 9, no. 1, 2019.

[18] Google, "Draco," Google, 2016. [Online]. Available: https://github.com/google/draco. [Accessed 29 08 2019].

[19] R. Schnabel and R. Klein, "Octree-Based Point-Cloud Compression," in *Eurographics/IEEE VGTC Conference on Point-Based Graphics*, 2006.

[20] MPEG, "MPEG 3DG and Requirements - Call for Proposals for Point Cloud Compression V2," 2017.

[21] R. Verhack, T. Sikora, L. Lange, R. Jongebloed, G. Van Wallendael and P. Lambert, "Steered Mixture-of-Experts for Light Field Coding, Depth Estimation, and Processing," in *IEEE International Conference on Multimedia and Expo*, 2017.

[22] J. Du, Z. Zou, Y. Shi and D. Zhao, "Zero Latency: Real-Time Synchronization of BIM Data in Virtual Reality for Collaborative Decision-Making," *Automation in Construction,* vol. 85, 2018.

[23] M. Hosseini, "Adaptive Rate Allocation for View-Aware Point-Cloud Streaming," University of Illinois, 2017.

[24] J. Park, P. A. Chou and J. Hwang, "Rate-Utility Optimized Streaming of Volumetric Media for Augmented Reality," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems,* vol. 9, no. 1, pp. 149-162, 2019.

[25] Internet Engineering Task Force, "Identifying Our Deliverables," 2018. [Online]. Available: https://mailarchive.ietf.org/arch/msg/quic/RLRs4nB1lwFCZ_7k0iuz0ZBa35s. [Accessed 23 08 2019].

[26] Y. Cui, Y. Li, C. Liu, X. Wang and M. Kühlewind, "Innovating Transport with QUIC: Design Approaches and Research Challenges," *IEEE Internet Computing,* vol. 21, no. 2, 2017.

[27] S. Petrangeli, D. Pauwels, J. van der Hooft, M. Ziak, J. Slowack, T. Wauters and F. De Turck, "A Scalable WebRTC-Based Framework for Remote Video Collaboration Applications," *Multimedia Tools and Applications,* vol. 78, no. 6, 2019.

[28] D. You, T. V. Doan, R. Torre, M. Mehrabi, A. Kropp, V. Nguyen, H. Salah, G. T. Nguyen and F. H. P. Fitzek, "Fog Computing as an Enabler for Immersive Media: Service Scenarios and Research Opportunities," *IEEE Access,* vol. 7, 2019.

[29] E. Bastug, M. Bennis, M. Medard and M. Debbah, "Toward Interconnected Virtual Reality: Opportunities, Challenges, and Enablers," *IEEE Communications Magazine,* vol. 55, no. 6, 2017.

[30] D. Szabo, A. Gulyas, F. H. P. Fitzek and D. E. Lucani, "Towards the Tactile Internet: Decreasing Communication Latency with Network Coding and Software Defined Networking," in *European Wireless Conference*, 2015.

[31] A. Javaheri, C. Brites, F. Pereira and J. Ascenso, "Subjective and Objective Quality Evaluation of 3D Point Cloud Denoising Algorithms," in *IEEE International Conference on Multimedia & Expo Workshops*, 2017.

[32] I. Viola, M. Rerábek, T. Bruylants, P. Schelkens, F. Pereira and T. Ebrahimi, "Objective and Subjective Evalation of Light Field Image Compression Algorithms," in *Picture Coding Symposium*, 2016.

Jeroen van der Hooft (S'14, M'18) obtained his M.Sc. and Ph.D. degrees in Computer Science Engineering from Ghent University, Belgium, in 2014 and 2019, respectively. He is currently active as a postdoctoral fellow at the Department of Information Technology, Ghent University – imec. His research interests include the end-to-end QoE optimization in adaptive video streaming and low-latency delivery of immersive video content.

Maria Torres Vega (S'14, M'17) obtained her M.Sc. degree in Telecommunication Engineering from the Polytechnic University of Madrid, Spain, in 2009, and her Ph.D. degree from the Eindhoven University of Technology, The Netherlands, in 2017. She is currently active as a postdoctoral fellow at Ghent University - imec. Her research interests include QoS and QoE in immersive multimedia systems and autonomous network management.

Tim Wauters (M'07) obtained his M.Sc. and Ph.D. degrees in Electrotechnical Engineering from Ghent University, Belgium, in 2001 and 2007, respectively. He is currently active as a postdoctoral fellow at Ghent University - imec. His work has been published in over 100 scientific publications in international journals and in the proceedings of international conferences. His research interests include network and service architectures, and management solutions for multimedia delivery.

Hemanth Kumar Ravuri obtained his B.Sc. degree in Electronics and Communication Engineering from the Jawaharlal Nehru Technological University, India, in 2014 and M.Sc. degree in Electrical Engineering with emphasis on Telecommunication Systems from BTH, Karlskrona, Sweden, in 2016. He is currently pursuing his Ph.D. degree at Ghent University - imec. His research interests include network and service architectures, next generation multimedia delivery, QoS and QoE.

Christian Timmerer (M'08, SM'16) obtained his M.Sc. and Ph.D. degrees from the Alpen-Adria-Universität Klagenfurt, Austria, in 2003 and 2006, respectively. He is currently active as an associate professor and vice-chair at the Institute of Information Technology at the same university. His research interests include immersive multimedia communication, streaming, adaptation, QoE and sensory experience. In 2013 he co-founded Bitmovin, where he is active as Chief Innovation Officer.

Hermann Hellwagner (S'85, A'88, M'95, SM'11) obtained his M.Sc. and Ph.D. degrees from the University of Linz, Austria, in 1983 and 1988, respectively. He is currently active as full professor at the Institute of Information Technology at the Alpen-Adria-Universität Klagenfurt. His research interests include multimedia communication and content adaptation, information-centric networking, and performance analysis of computer and communication systems.

Filip De Turck (S'95, M'98, SM'12) obtained his M.Sc. and Ph.D. degrees in Electronic Engineering from Ghent University, Belgium, in 1997 and 2002, respectively. He is currently active as a full professor at Ghent University - imec, where he leads the network and service management research group at the Department of Information Technology. His research interests include telecommunication network and service management, and design of efficient virtualized network systems.

MMTC OFFICERS (Term 2018 — 2020)

CHAIR

Honggang Wang
UMass Dartmouth
USA

STEERING COMMITTEE CHAIR

Sanjeev Mehrotra
Microsoft
USA

VICE CHAIRS

Pradeep K Atrey (North America)
Univ. at Albany, State Univ. of New York
USA

Wanqing Li (Asia)
University of Wollongong
Australia

Lingfen Sun (Europe)
University of Plymouth
UK

Jun Wu (Letters&Member Communications)
Tongji University
China

SECRETARY

Shaoen Wu
Ball State University
USA

STANDARDS LIAISON

Guosen Yue
Huawei
USA

MMTC Communication-Frontier BOARD MEMBERS (Term 2016—2018)

Dalei Wu	Director	University of Tennessee at Chattanooga	USA
Danda Rawat	Co-Director	Howard University	USA
Melike Erol-Kantarci	Co-Director	University of Ottawa	Canada
Kan Zheng	Co-Director	Beijing University of Posts & Telecommunications	China
Rui Wang	Co-Director	Tongji University	China
Lei Chen	Editor	Georgia Southern University	USA
Tasos Dagiuklas	Editor	London South Bank University	UK
ShuaiShuai Guo	Editor	King Abdullah University of Science and Technology	Saudi Arabia
Kejie Lu	Editor	University of Puerto Rico at Mayagüez	Puerto Rico
Nathalie Mitton	Editor	Inria Lille-Nord Europe	France
Zheng Chang	Editor	University of Jyväskylä	Finland
Dapeng Wu	Editor	Chongqing University of Posts & Telecommunications	China
Luca Foschini	Editor	University of Bologna	Italy
Mohamed Faten Zhani	Editor	l'École de Technologie Supérieure (ÉTS)	Canada
Armir Bujari	Editor	University of Padua	Italy
Kuan Zhang	Editor	University of Nebraska-Lincoln	USA