# Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

March 2023 Volume XV Number 1 ISSN 2061-2079

*MESSAGE FROM THE EDITOR-IN-CHIEF*

Recent advances on high performance computing, mobile networking, and security ..... *Pal Varga* 1

*PAPERS FROM OPEN CALL*

Novel Metasurface based Microstrip Antenna Design for for Gain Enhancement RF Harvesting ..... *Hayder Almizan, Marwah Haleem Jwair, Yahiea Al Naiemy, Zaid A. Abdul Hassain, Lajos Nagy, and Taha A. Elwi* 2

On the Convex Hull of the Achievable Capacity Region of the Two User FDM OMA Downlink ..... *Zoltán Belső and László Pap* 9

Enhancing Visual Domain Randomization with Real Images for Sim-to-Real Transfer ..... *András Béres and Bálint Gyires-Tóth* 15

Machine Learning Use-Cases in C-ITS Applications ..... *Norman Bereczki and Vilmos Simon* 26

Closed-loop Orchestration for Cloud-native Mobile IPv6 ..... *Ákos Leiter, Edina Lami, Attila Hegyi, József Varga, and László Bokor* 44

Optimizing the Performance of the Iptables Stateful NAT44 Solutions ..... *Gábor Lencse, Keiichi Shimi* 55

BER-Aware Backscattering Design for Energy Maximization at RFID Passive Tag .... *Basma Mostafa, Miklos Molnar, Mohamed Saleh, Abderrahim Benslimane, and Sally Kassem* 64

Application-Aware Analysis of Network Neutrality: A Scalable Real-Time Method ..... *Péter Orosz, Tamás Skopkó, Tamás Marosits* 77

Techno-economic analysis on Mobile Network Sharing contribution to social welfare at 4G-5G area in Hungary ..... *Gábor Földes* 87

*CALL FOR PAPER / PARTICIPATION*

MeditCom 2023 / IEEE International Mediterranean Conference on Communications and Networking IEEE MeditCom 2023, Dubrovnik, Croatia ..... 98

GLOBECOM 2023 / IEEE Global Communications Conference IEEE GLOBECOM 2023, Kuala Lumpur, Malaysia ..... 99

Advanced Distributed Optimization for Edge-Intelligence Communication Systems Special Issue ..... 101

*ADDITIONAL*

Guidelines for our Authors ..... 100

Technically Co-Sponsored by

IEEE ComSoc IEEE Communications Society

hte

IEEE HUNGARY SECTION

## Indexing information

Infocommunications Journal is covered by Inspec, Compendex and Scopus.
**Infocommunications Journal is also included in the Thomson Reuters – Web of ScienceTM Core Collection, Emerging Sources Citation Index (ESCI)**


## Infocommunications Journal

Technically co-sponsored by IEEE Communications Society and IEEE Hungary Section

**Supporters**

FERENC VÁGUJHELYI – president, Scientific Association for Infocommunications (HTE)

**Editorial Office** (Subscription and Advertisements)**:**
Scientific Association for Infocommunications
H-1051 Budapest, Bajcsy-Zsilinszky str. 12, Room: 502
Phone: +36 1 353 1027
E-mail: info@hte.hu • Web: www.hte.hu

**Articles can be sent also to the following address:**
Budapest University of Technology and Economics
Department of Telecommunications and Media Informatics
Phone: +36 1 463 4189, Fax: +36 1 463 3108
E-mail: pvarga@tmit.bme.hu

**Subscription rates for foreign subscribers:** 4 issues 10.000 HUF + postage

Publisher: PÉTER NAGY

HU ISSN 2061-2079 • Layout: PLAZMA DS • Printed by: FOM Media

# Recent advances on high performance computing, mobile networking, and security

Pal Varga

WITH the advent of new generation large language model-based tools, such as ChatGPT, many worktasks and positions must re-evaluate themselves. This applies to writing editorial messages as well. It is no secret: I am using Google for searching scientific articles. Similarly: I used ChatGPT to extract some key information of the articles in this issue. The result needed validation and verification (just like Google search results), and some re-writing from my side – but the world has changed. Hopefully for the better, even for the long term! So, let us see what are the articles about!

The paper by Mostafa et. al. presents an enhancement in radio frequency (RF) harvesting for a conventional patch antenna using a metasurface layer (MSL) consisting of a modified Jerusalem cross. RF harvesting measurements showed a significant enhancement in the output voltage when the MSL was involved, and the proposed antenna realizes an excellent symmetry in the radiation pattern after the proposed MSL introduction.

Zoltán Belső and László Pap address the problem of sharing available resources (such as bandwidth and power) in a multiple access channel system in order to determine how to allocate those resources between two users to achieve the highest achievable capacity region. An important achievement here is the formulation and derivation of the achievable capacity region for a downlink communication system using frequency division multiple access (FDMA) scheme with two users.

András Béres and Bálint Gyires-Tóth propose a method to enhance the performance of reinforcement learning agents through sim-to-real transfer using real-world images alongside visual domain randomization in the training procedure. They achieve this by training variational autoencoders (VAE) on both real and simulated frames, and then using the representations produced by the encoders to train reinforcement learning agents. The proposed method is evaluated against a variety of baselines and is shown to improve the learned representation effectiveness and robustness.

Norman Bereczki and Vilmos Simon provide a structured and comprehensive overview of machine learning (ML) usecases in the field of Cooperative Intelligent Transportation Systems (C-ITS). The authors categorized C-ITS applications based on the level of cooperativeness and provided a summary of how machine learning contributes to each layer of smart transportation. The paper shows how C-ITS applications are built on vehicle-level, inter vehicle-level and infrastructure level, and what is the impact of ML in these cases.

In their article, Leiter et. al. aim to show how failover and scaling can be applied to cloud-native Mobile IPv6 (CN-MIPv6) using closed-loop orchestration on top of the Open Network Automation Platform. They present numerical results and calculations on the utilization of failover, scaling, and availability in the case of cloud-native IP-level mobility management – findings, that can be applied to any (V)NF.

Gábor Lencse and Keiichi Shima explores the performance of iptables in stateful NAT44 translation and investigates the effect of various settings and parameters on its performance. The authors compare iptables to nftables, and conclude that iptables is still appropriate for implementing CGN due to its superior performance in handling a high number of connections. The authors also propose a new benchmarking methodology for stateful NATxy gateways to quantify the connection setup and connection teardown performance.

In their article, Mostafa et.al. propose a proactive, dynamic, and distributed network monitoring mechanism with monitor placement and scheduling for 6LoWPAN-based IoT networks intended for mission-critical applications. The results show that the proposed model achieves full network coverage while minimizing energy consumption, computational and communication overhead, and preserving scarce resources by effectively distributing the monitoring role.

Péter Orosz, Tamás Skopkó and Tamás Marosits presents a real-time, application-aware method for assessing network neutrality. This aims to provide a user-centric analysis of potential restraints affecting internet access quality and introduces three novelties to support this goal: application-specific measurements involving real content and traffic, measured traffic originating from the content provider's cloud infrastructure, and reference created in real-time. The proposed method is validated through a proof-of-concept use case of video streaming from a public VoD provider and laboratory measurements.

In his paper Gábor Földes analyzes the impact of mobile network sharing on social welfare in Hungary, specifically in the context of 4G-5G mobile broadband rollout. He argues that network sharing can be a cost efficiency enabler, accelerating price decreases, improving coverage and capacity, and ultimately contributing to social welfare increase. Mobile network sharing can benefit the telco sector and contribute to social welfare, but further research is needed to assess the appropriate level of access pricing to shared infrastructure and the passthrough of efficiency gains to customers.

With this overview, we hope you will enjoy the Infocommunications Journal papers in the first issue of 2023.

**Pal Varga** is the Head of Department of Telecommunications and Media Informatics at the Budapest University of Technology and Economics. His main research interests include communication systems, Cyber-Physical Systems and Industrial Internet of Things, network traffic analysis, end-to-end QoS and SLA issues – for which he is keen to apply hardware acceleration and artificial intelligence, machine learning techniques as well. Besides being a member of HTE, he is a senior member of IEEE, where he is active both in the IEEE ComSoc (Communication Society) and IEEE IES (Industrial Electronics Society) communities. He is Editorial Board member in many journals, and the Editor-in-Chief of the Infocommunications Journal.

# Novel Metasurface based Microstrip Antenna Design for Gain Enhancement RF Harvesting

Hayder Almizan[1], Marwah Haleem Jwair[2], Yahiea Al Naiemy[3], Zaid A. Abdul Hassain[4], Lajos Nagy[3], and Taha A. Elwi[5, *]

***Abstract*—This paper presents an enhancement in radio frequency (RF) harvesting for conventional patch antenna using a metasurface layer (MSL). The key point behind such enhancement is inspired by Friis' equation which states; increasing the antenna gain leads to an increase in the received power. To achieve this goal, a MSL consists of 5×5-unit cells of a modified Jerusalem cross are proposed. The proposed MSL provides gain enhancement of about 10 dBi while the gain of the patch antenna is about 1 dBi. The proposed MSL is fabricated, compacted to the antenna and experimentally characterized. The empirical results indict an excellent agreement with the numerical results in terms of |S11| and radiation patterns. In addition, a set of RF harvesting measurements are made for patch antenna with and without the MSL. The comparison between measurements shows a significant enhancement in the output voltage when the MSL is involved.**

***Index Terms*—MTL, gain control, RF harvesting, Friis.**

## I. INTRODUCTION

Radio frequency (RF) harvesting is conceptually an energy conversion method employed for converting energy from the electromagnetic field into the electrical domain (i.e., currents and voltages) [1]. Therefore, design of an RF harvesting system needs to interface between the electromagnetic fields and the electronics circuits.

RF energy harvesting was proposed to make use of the power from the ambient electromagnetic signals within the microwave range such as TV, GSM, and Wi-Fi signals [2].

Despite the fact the RF source has the lowest power density, the privilege of RF energy is the broadest availability compared with the solar radiation and infrared thermal energies in terms of weather, time, and location. This reason makes RF energy harvesting promises to produce energy meets the requirement of low-power budget sensors, devices, and systems [3].

The history of RF energy harvesting back to 1905, when Nikola Tesla (1856-1943) proposed wireless power transfer as a means of electrical energy transfer [4]. Later in the 1950s, an RF energy harvesting system was used for space microwave power-beaming applications and powering autonomous drones [5].

However, growing in low-power electronics systems make RF energy harvesting system desirable sub-system in some applications. Therefore, many researches have been conducted to meet the new require. As can be referred [6], multiple antennas have been proposed to increase the amount of energy derived from a given space by an energy-harvesting device. [7] presented a battery-free programmable sensing and computational platform for radio frequency identification (RFID) applications. [8] Proposed a high-efficiency harvester that can harvest low input RF power effectively. The researchers in [9] presented a design of monopole antenna with circular polarization for RF harvesting for wireless sensor node. The author of [10] utilized shorted ring slot antenna for circular polarization in GSM 900 band. In [11], circular polarization is formed via differentially-fed while the omnidirectional antenna is realized by compacting back-to-back patch antenna. Researchers of [12], the circular polarization was made via slotted-two circular-patch radiator with an electromagnetic coupling feed. A dipole antenna array with sequential rotation feeding was introduced in [13]. Researchers of [14] discussed a design and analysis of RF harvesting system consists of a microstrip patch antenna and three-stage Cockcroft-Walton rectifier.

In work [15], the authors proposed an efficient and low-cost RF energy harvester operating at 2.4 GHz. The proposed system comprises of a microstrip patch antenna, a filter, rectifier as well as a matching circuit. In [16], the researchers proposed a novel compact dual-circinal antenna with an omnidirectional characteristic for RF power transmission. Besides, a new approach to decrease the antenna size, was discussed. A patch of tapered-slit octagonal-shaped was employed to achieve the circular polarization as well as wide-beam in [17]. Authors of [18] proposed a hexagonal-shaped microstrip patch antenna with an objective to achieve an antenna with good radiation performance and better impedance matching. In [19], the authors proposed a novel broadband, wide-angle and back-to-back microstrip antenna for RF harvesting.

[1] Department of Electronics and Communications, College of Engineering, Kufa University, Najaf, Iraq,
[2] Department of Information and Communication Engineering, College of information engineering, Al-Nahrain University, Baghdad, Iraq,
[3] Broadband Infocommunications and Electromagnetic Theory, University of Budapest University of Technology and Economics, Budapest, Hungary
[4] Electrical Engineering, College of Engineering, Mustansiriyah University, Baghdad, Iraq,
[5, *] International Applied and Theoretical, (E-mail: taelwi82@gmail.com)

DOI: 10.36244/ICJ.2023.1.1

On the other hand, MSL has been employed for RF harvesting by two paths of researches. The first path uses MSL to enhance the receiver antenna performance. As in [20], the authors presented this idea by design an electrically small rectenna at the global positioning system (GPS) L1 frequency (1.5754 GHz) with a metamaterial-inspired near-field resonant. The key point of [21] was to match the input impedance of the rectifying circuit. Researchers of [22] proposed a wireless power harvester based on a rectifier, and antenna added to hybrid frequency selective surface. The second path of researches applied MSL as a RF harvester directly instead of the receiver antenna. [23] Proposed a design of dual-polarization and multi-focus near-field focusing reflective MSL for wireless power transfer.

The second research path has been inspired by the idea of perfect electromagnetic absorber introduced in [24]. Authors of [25] present a design for absorbing metamaterial with near-unity absorbance where the proposed structure consists of two resonators of metamaterial that couple separately to electric and magnetic fields. In [26], a design of MSL with polarization-insensitive characteristics and wide-angle of the reception was proposed for electromagnetic energy harvesting. Authors of [27] proposed an electromagnetic energy harvester based MSL array of 9×9 pixelated unit cells with dual-band and polarization-independent. The fundamental parts as a block diagram of the RF harvesting system are shown in Figure 1.

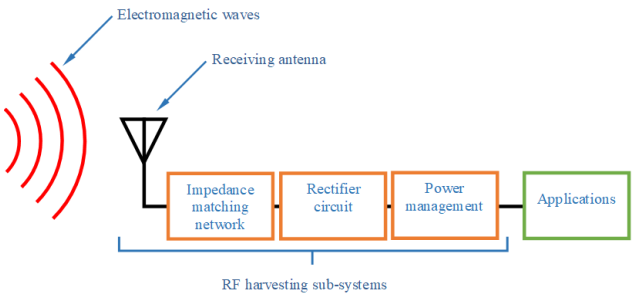



Figure 1; Fundamental parts of RF harvesting system.

The first part represents the receiving antenna (or the harvester antenna), where this part is responsible for converting the incident electromagnetic waves into an electrical signal. The second part is used to enhance harvesting via an impedance matching network. The third part is a rectifier circuit. Lastly, the produced power from the previous stage is managed by a power management sub-system.

This work is concerned with the first sub-system (receiving antenna) of the RF harvester and how to improve RF energy harvesting via improving the characteristics of this part. The second level of RF harvesting enhancement is based on the equation of Friis for the received power $P_r$ [28]

$$P_r = P_t + G_t + G_r + 20\log\left(\frac{\lambda}{4\pi R}\right) \quad (1)$$

where $P_t$ is the transmitted power in (dB), $G_t$ is the transmitter gain in (dBi), $G_r$ is the receiver (or the harvesting antenna) gain in (dBi), $\lambda$ is the wavelength and $R$ is the distance between the proposed system and the transmitter in meters. Equation (1) clearly indicates that; increasing $G_r$ leads to an increase the received power. In this work, a certain MSL will be employed to enhance the received antenna gain to enhance harvested power $P_r$.

## II. ANTENNA DESIGN AND STRUCTURE

Figure 2 shows the design of the proposed microstrip antenna where the patch antenna is directly fed by a coaxial probe structure with a 50 Ω SMA port and located at $(x = 0, y = -4.5, z = 0)$ mm. The patch geometry is inspired from [29] based on truncated rectangular structure. As well as, the patch is etched to create four slots. The reason for using this technique is to enhance the antenna matching impedance below -10 dB, however, the radiation pattern symmetry could be ruined [30]. The antenna substrate is made of FR4 material with $\varepsilon_r = 4.3, \tan\delta = 0.0025$, and thickness of 2 mm. The substrate is designed to shape as a square of 280 mm side length. The details antenna dimensions are listed in Table 1.

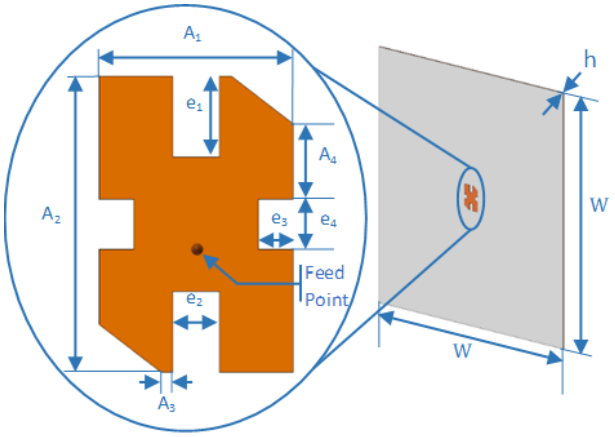



Figure 2; The proposed microstrip antenna structure.

TABLE I
THE PROPOSED MICROSTRIP ANTENNA DIMENSIONS.

| Parameter | Dimension (mm) |
|---|---|
| W | 280 |
| h | 2 |
| $A_1$ | 19.31 |
| $A_2$ | 27.43 |
| $A_3$ | 1.17 |
| $A_4$ | 6.98 |
| $e_1$ | 7.51 |
| $e_2$ | 4.66 |
| $e_3$ | 3.44 |
| $e_4$ | 4.65 |

The proposed antenna reflection coefficient and radiation pattern are clarified in Figure 3 for both simulated and measured results. The $|S_{11}|$ resonates at 2.47 GHz with about 1 dBi gain as seen in Figures 3(b)-(c). Besides, both simulated and measured results are splendid matched.

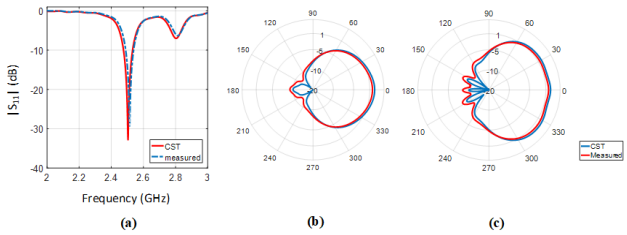



Figure 3; The $|S_{11}|$ spectra and the antenna radiation pattern based simulated and measured results for the proposed antenna: (a) the $|S_{11}|$ spectra, (b), and (c) the antenna radiation pattern at $\varphi = 0^{\circ}$ and $\varphi = 90^{\circ}$ respectively.

## III. THE PROPOSED MSL

The proposed MSL is an array of 5×5 unit cells. Each unit cell is constructed from Jerusalem cross-section plus four L-shape strips as seen in Figure 4. The unit cell dimensions are utilized to 60 mm, which is about λ/2. Choosing such dimensions is to avoid forming grating lobes [6]. The detailed dimensions of the proposed unit cell are clarified in Table 2.

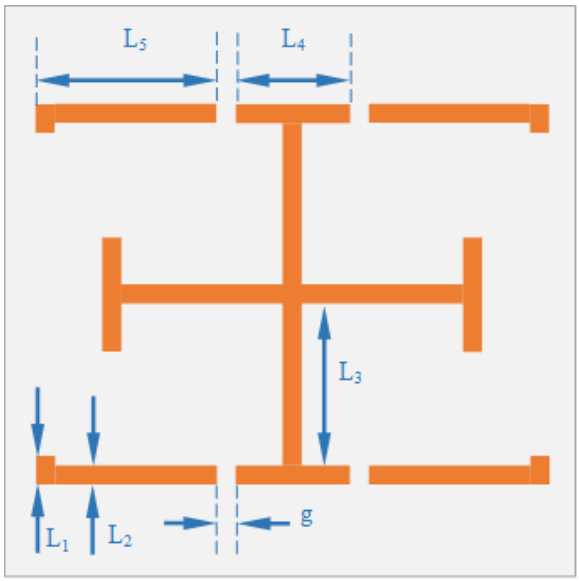



Figure 4; Unit cell structure.

TABLE II
UNIT CELL DIMENSIONS.

| Parameter | Dimension (mm) |
|---|---|
| $L_1$ | 3 |
| $L_2$ | 2 |
| $L_3$ | 17 |
| $L_4$ | 12 |
| $L_5$ | 19 |
| G | 2 |

The proposed unit cell electromagnetic characterizations are tested numerically using CST MWS [31]. The proposed unit cell is positioned at the centre of a virtual waveguide as shown in Figure 5(a) to test the electromagnetic performance. The boundary conditions are selected as: the top and bottom sides (perpendicular on y-axis) are assigned as a Perfect Magnetic Conductors (PMC) while, the left and right sides (perpendicular on *x*-axis) are assigned as a Perfect Electric Conductors (PEC). TEM-modes are excited via two ports along z-axis as shown in Figure 5(a). For this, Figure 5(b) shows the magnitude of $S_{12}$ for the two cases ON and OFF. It is important to mention that the authors decided to realize a resonant frequency for the unit cell at Logic-1 to 2.67 GHz for the design specifications [15], while, in case of Logic-0, the frequency resonance is completely disappeared from the frequency band of interest. Consequently, the maximum power transfer can be obtained at logic-1 only, while, the power transfer would be shut off at logic-0. The corresponding phase of the forward transmission coefficient $S_{21}$ is plotted in Figure 5(c) for both statues ON and OFF. For statue-ON, the matching impedance occurs at the resonant frequency 2.67 GHz where the phase is zero degree. This phenomenon declares that the impedance's imaginary part is vanished and it confirms that the result of maximum power transfer can be obtained at logic-1.

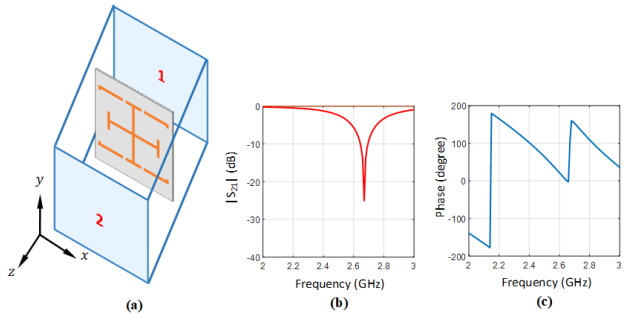



Figure 5; Unit cell performance characterizations: (a) Numerical setup, (b) the magnitude of $S_{21}$, and (c) phase performance.

In another aspect, Figure 6 shows the surface current distributions of the proposed MSL unit cell at phase $= 90^{\circ}$ of the incident wave. In general, it is found that the proposed MSL unit cell shows asymmetrical surface current distribution along the *x*- and *y*-planes due to the asymmetrical unit cell geometry along *x*- and *y*-planes. This realizes the phenomena of astigmatism affects, which effects on the radiation patterns, as will be seen later. Nevertheless, the surface current distribution is reached around 29 dBA/*m* at 2.65 GHz. This enhancement in the surface current distribution is achieved by the resonance at 2.45 GHz; which removes the reactive impedance parts to realize the maximum power transfer radiation [31].

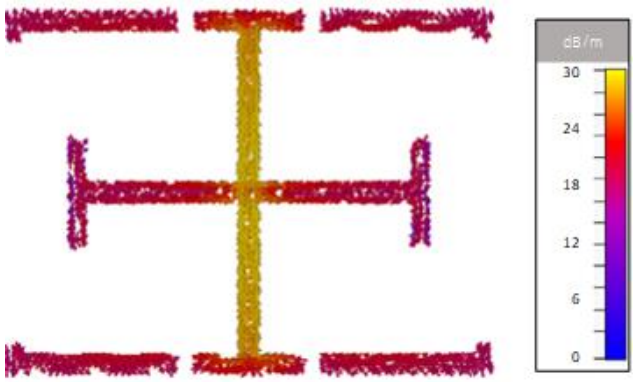



Figure 6; Surface current distributions of the proposed unit cell at 2.65 GHz.

## IV. MSL LAYER DESIGN AND STRUCTURE

The proposed MSL layer is based on an array of $5 \times 5$-unit cells. To realize the configuration purpose, the MSL layer constructed from the proposed unit cell is mounted on a flexible thin layer of 0.1 $mm$ thickness, FR4 substrate. The overall individual unit cell dimensions are $60\ mm \times 60\ mm$, where the physical dimensions of the unit cell are $54\ mm \times 40\ mm$ as well as spaces between neighbored unit cells are $3\ mm$ on the $x$-axis and $10\ mm$ on $y$-axis. To ensure minimum coupling between unit cells, the periodicity of the unit cell is adjusted to be $60\ mm$ (~$\lambda/2$) [32]. Figure 7 shows the proposed MSL layer compacted to the microstrip antenna.

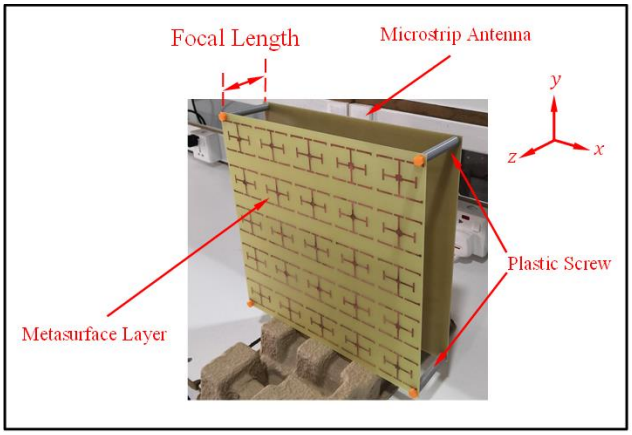



Figure 7; The proposed MSL layer with the antenna structure.

CST MWS is employed to study the best MSL location and array size with respect to the patch antenna. By changing the MSL array configuration size from 1×1, 3×3, and 5×5 and the patch antenna-MSL distance, the antenna gain is found to be significantly changing, as seen in Figure 8. To achieve 10 dBi gain enhancement, the MSL must consist of 5×5 unit cells and located at 70 mm in front of the patch antenna.

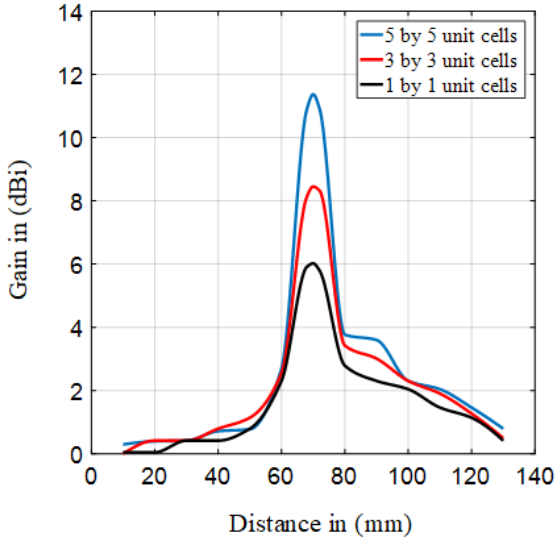



Figure 8; A parametric study for the proposed MSL array size performance.

In another aspect, Figure 9 shows $S_{11}$ measurements of the proposed MSL systems with 2D radiation patterns at both E- and H-planes. In general, it is found that the proposed MSL layer shows symmetrical radiation patterns at both E- and H-planes in the boresight. This realizes the phenomena of the astigmatism solving by conducting the use of the proposed MSL array.

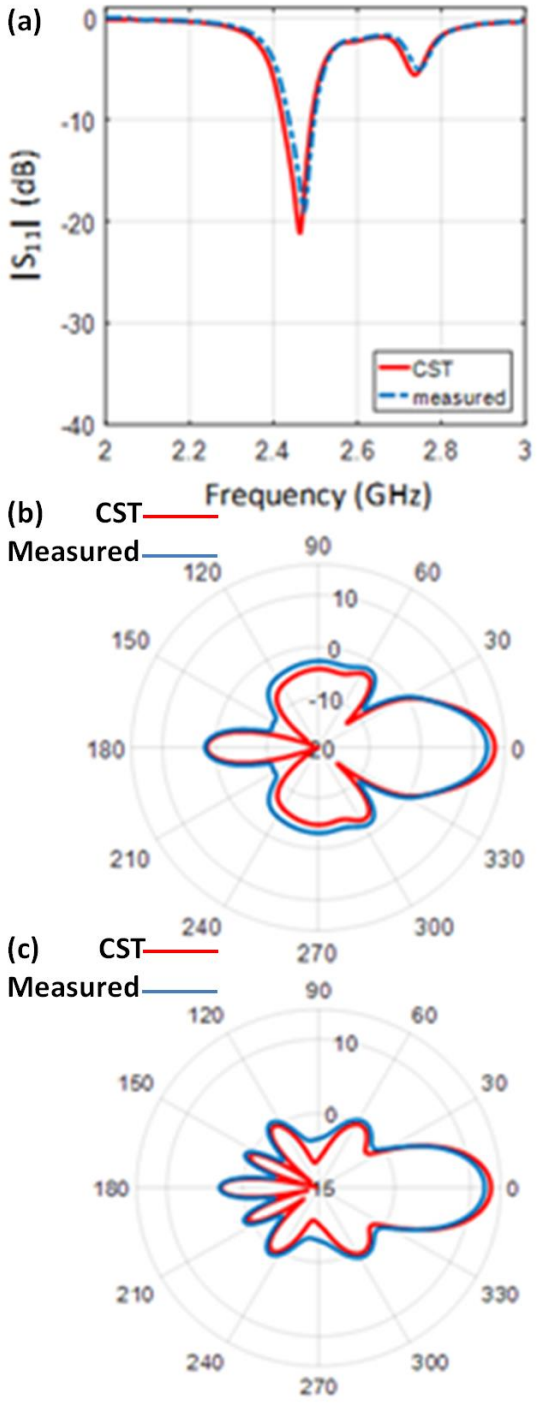



Figure 9; The proposed antenna systems measurements: (a) S11, (b) E-plane, and (c) H-plane.

## V. HARVESTING MEASUREMENTS

RF harvesting measurements are conducted for the patch antenna and repeated for the patch antenna loaded MSL. Both measurements cases are submitted to the similar scenario where a transmitter fed by a voltage control oscillator (VCO) is placed at 0.5m, 1.0m, 1.5m, and 2.0m away from the proposed antenna based RF harvesting process is shown in Figure 10. The same transmitter is involved for all measurements. Hence, the transmitter gain does not affect the measurements.

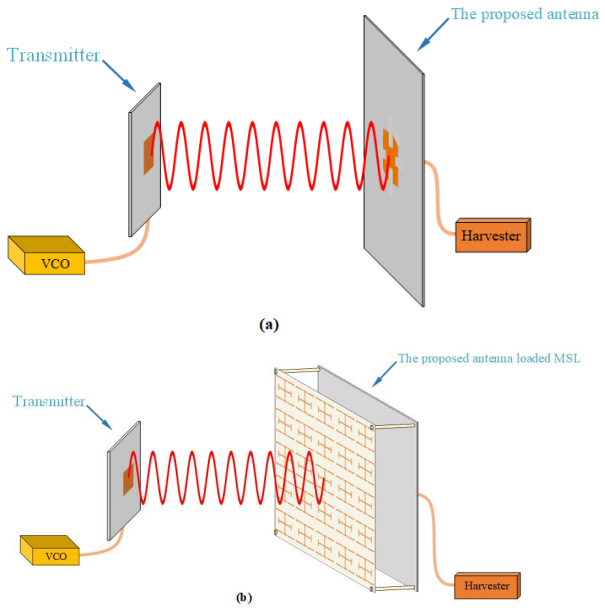



Figure 10; RF harvesting measurements: (a) The proposed antenna and (b) The proposed antenna loaded MSL.

The output DC voltage values from the RF harvester are described in Figure 11 when the load is 1 kΩ at 2.4, 2.45, and 2.5 GHz. The recorded measurements indicate two distinguished points. First, the output DC voltage levels decay generally when the separation distance between the transmitter and the proposed antenna increase. This behavior confirms the harmonic between measurements and the power inverse-square law. Second, the output DC voltages at 2.45 GHz when the patch antenna with MSL is found to be higher than other recorded results. This is due to the fact of MSL characterizes as a narrow bandwidth material.

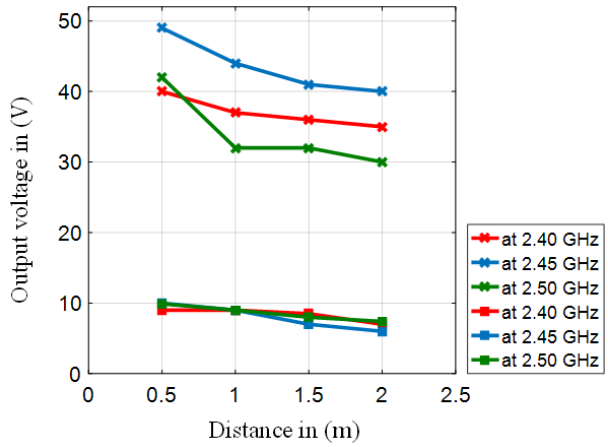



Figure 11; The output DC voltage from the RF harvester at 1 KΩ.

## VI. CONCLUSIONS

RF harvesting measurements show distinguished improvement in the received power when the proposed patch antenna is loaded by MSL. The proposed MSL is designed as array of 5×5-unit cells of a modified Jerusalem cross. The MSL provides gain enhancement of 10 dBi. The empirical results of the proposed antenna-MSL indicate excellent agreement respecting RF harvesting. It is found that the proposed antenna realizes an excellent symmetry in the radiation pattern after the proposed MSL introduction. The proposed provides an excellent enhancement in the harvested output DC voltage due to the introduction of the proposed MSL structure.

## REFERENCES

[1] Sabah Hassan Ghadeer, Sharul Kamal Abd. Rahim, Mohammad Alibakhshikenari, Bal S. Virdee, Taha A. Elwi, Amjad Iqbal, Muath Al-Hasan, "An innovative fractal monopole MIMO antenna for modern 5G applications", AEU - International Journal of Electronics and Communications, Volume 159, 2023, 154480, ISSN 1434-8411, DOI: 10.1016/j.aeue.2022.154480.

[2] Al-khaylani HH, Elwi TA, Ibrahim AA. Optically Remote Control of Miniaturized 3D Reconfigurable CRLH Printed Self-Powered MIMO Antenna Array for 5G Applications. Micromachines. 2022; 13(12):2061. DOI: 10.3390/mi13122061.

[3] Al-khaylani, HH, Elwi, TA, Ibrahim, AA. Optically remote-controlled miniaturized 3D reconfigurable CRLH-printed MIMO antenna array for 5G applications. Microw Opt Technol Lett. 2022; 1- 8. DOI: 10.1002/mop.33504.

[4] Bashar S. Bashar; Z.A. Rhazali; Taha A. Elwi; Halina Misran; Marwa M. Ismail; Maki Mahdi, and Ali Ihsan Alanssari,"Antenna Beam forming Technology Based Enhanced Metamaterial Superstrates," 2022 IEEE 3rd KhPI Week on Advanced Technology (KhPIWeek), 2022, pp. 1-5, DOI: 10.1109/KhPIWeek57572.2022.9916458.

[5] M. Manteghi, "A Wideband Electrically Small Transient-State Antenna," in IEEE Transactions on Antennas and Propagation, vol. 64, no. 4, pp. 1201-1208, April 2016, DOI: 10.1109/TAP.2016.2525828.

[6] G. Kiani, and T. Bird, "ASK modulator based on switchable FSS for THz applications," RADIO SCIENCE, vol. 46, no. 2, pp. 1-8, Apr. 2011, DOI: 10.1029/2010RS004465.

[7] S. Henthorn, K. L. Ford and T. O'Farrell, "Frequency selective surface loaded antenna for direct antenna modulation," 2017 11th European Conference on Antennas and Propagation (EUCAP), 2017, pp. 731-734, DOI: 10.23919/EuCAP.2017.7928325.

[8] Alaukally, MNN, Elwi, TA, Atilla, DC. Miniaturized flexible metamaterial antenna of circularly polarized high gain-bandwidth product for radio frequency energy harvesting. Int J Commun Syst. 2022; 35( 3):e5024. DOI: 10.1002/dac.5024.

[9] M. M. Ismail, T. A. Elwi and A. J. Salim, "Design and Simulation of a CRLH Transmission Line Antenna of a Hilbert Fractal Geometry for S-Band Appications," 2021 International Conference on Electrical, Computer and Energy Technologies (ICECET), 2021, pp. 1-5, DOI: 10.1109/ICECET52533.2021.9698449.

[10] A. Babakhani, D. B. Rutledge and A. Hajimiri, "Transmitter Architectures Based on Near-Field Direct Antenna Modulation," in IEEE Journal of Solid-State Circuits, vol. 43, no. 12, pp. 2674-2692, Dec. 2008, DOI: 10.1109/JSSC.2008.2004864.

[11] Y. Al-Naiemy, T. A. Elwi, and N. Lajos, "Enhancing the Microstrip Antenna Gain Using a Novel FSS Lens Based on a Single Layer", Proceedings of the 3rd Colloquium on Microwave Communication, Aug. 2018. DOI: 10.1109/CSNDSP.2018.8471786.

[12] Y. Alnaiemy, T. A. Elwi, N. Lajos, and T. Zwick, "A Systematic Analysis and Design of a High Gain Microstrip Antenna based on a Single FSS Layer," IEEE INFOCOMMUNICATIONS JOURNAL, vol. 10, pp. 1-9, December, 2018. DOI: 10.36244/ICJ.2018.4.4.

[13] T. A. Elwi, H. M. Al-Rizzo, N. Bouaynaya, M. M. Hammood, and Y. Al-Naiemy, "Theory of gain enhancement of UC–PBG antenna structures without invoking Maxwell's equations: an array signal processing approach," Progress In Electromagnetics Research B, vol. 34, pp. 15-30, August 2011. DOI: 10.2528/PIERB11062709.

[14] Elwi, T.A., Abdul Hassain, Z.A. and Tawfeeq, O.A. (2019), Hilbert metamaterial printed antenna based on organic substrates for energy harvesting. IET Microw. Antennas Propag., 13: 2185-2192. DOI: 10.1049/iet-map.2018.5948.

[15] R. V. Sravya and R. Kumari, "Gain enhancement of patch antenna using L-slotted mushroom FSS," 2018 Conference on Signal Processing and Communication Engineering Systems (SPACES), Vijayawada, 2018. DOI: 10.1109/ISANP.2017.8228987.

[16] Yongxing Che, Xinyu Hou and Peng Zhang, "Design of multiple FSS screens with dissimilar periodicities for directivity enhancement of a dual-band patch antenna," Proceedings of the 9th International Symposium on Antennas, Propagation and EM Theory, Guangzhou, 2010. DOI: 10.1109/ISAPE.2010.5696464.

[17] A. Pirhadi, H. Bahrami and J. Nasri, "Wideband High Directive Aperture Coupled Microstrip Antenna Design by Using a FSS Superstrate Layer," IEEE Trans. Antennas Propag., vol. 60, no. 4, pp. 2101-2106, Apr., 2012. DOI: 10.1109/TAP.2012.2186230.

[18] Kodera, "Smart metasurface with non-reciprocity for Fog layer in IoT environment," IEEE International Symposium on Electromagnetic Compatibility and 2018 IEEE Asia-Pacific Symposium on Electromagnetic Compatibility (EMC/APEMC), Singapore, pp. 912-914, 2018. DOI: 10.1109/ISEMC.2018.8393914.

[19] L. Liu, L. Kang, T. S. Mayer, and D. H. Werner, "Hybrid metamaterials for electrically triggered multifunctional control," Nature Comm., vol. 7, no. 1, 2015. DOI: 10.1038/ncomms13236.

[20] P. Hosseini, C. D. Wright, and H. Bhaskaran, "An optoelectronic framework enabled by low-dimensional phase-change films," Nature, vol. 511, no. 7508, pp. 206–211, 2014. DOI: 10.1038/nature13487.

[21] Haran ManoharanG Dilip KumarA Ferris GarvinRamesh .SRamesh .S, "HEXAGONAL MICROSTRIP PATCH ANTENNA FOR EARLY STAGE SKIN CANCER IDENTIFICATION" Vol. 79, Issue 7, 2020, pp. 555-566. DOI: 10.1615/TelecomRadEng.v79.i7.20.

[22] T. A. Elwi, D. A. Jassim, H. H. Mohammed, "Novel miniaturized folded UWB microstrip antenna-based metamaterial for RF energy harvesting," International Journal of Communication Systems, Vol. 1, issue 2, January 2020. DOI: 10.1002/dac.4305.

[23] Z. Al-Dulaimi, T. A. Elwi, and D. C. Atilla, "Design of a Meander Line Monopole Antenna Array Based Hilbert-Shaped Reject Band Structure for MIMO Applications," IETE Journal of Research, Vol. 66, issue 1, pp. 1-10, Mar., 2020. DOI: 10.1080/03772063.2020.1743207.

[24] G. Kizer, "Digital Microwave Communication Engineering Point-to-Point Microwave Systems," New Jersey: John Wiley & Sons, 2013. DOI: 10.1002/9781118636336.ch16.

[25] Ahmed I. Imran, Taha A. Elwi, A cylindrical wideband slotted patch antenna loaded with Frequency Selective Surface for MRI applications, Engineering Science and Technology, an International Journal, Volume 20, Issue 3, 2017, Pages 990-996, ISSN 2215-0986, DOI: 10.1016/j.jestch.2017.04.001.

[26] F. S. Barnes, and B. Greenebaum, "Bioengineering and Biophysical Aspects of Electromagnetic Fields", 3rd ed., Taylor & Francis Group, Boca Raton, 2007. DOI: 10.1201/9781315186580.

[27] CST Microwave Studio, [online] Available at: http://www.cst.com [Accessed: 28 May 2018].

[28] T. A. Elwi, "Printed Microwave Metamaterial-Antenna Circuitries on Nickel Oxide Polymerized Palm Fiber Substrates," Nature Scientific Reports, vol. 9, number 2174, pp. 1-14, Jan. 2019. DOI: 10.1038/s41598-019-39736-8.

[29] Taha A. Elwi, Novel UWB printed metamaterial microstrip antenna based organic substrates for RF-energy harvesting applications, AEU - International Journal of Electronics and Communications, Volume 101, 2019, pp. 44-53, DOI: 10.1016/j.aeue.2019.01.026.

[30] H. S. Ahmed and T. A. Elwi, "On the design of a reject band filter for antennas mutual coupling reduction", International Journal of RF and Microwave Computer-Aided Engineering, vol. 11, number 3, pp.1-11, April 2019. DOI: 10.1002/mmce.21797.

[31] H. M. Al-Sabbagh, T. A. Elwi, Y. Al-Naiemy, and H. M. Al-Rizzo, "A Compact Triple-Band Metamaterial-Inspired Antenna for Wearable Applications", Microwave and Optical Technology Lett., vol. 11, number 2, May 2019, DOI: 10.1002/mop.32067.

[32] Marwah Haleem and Taha A. Elwi, "Circularly Polarized Metamaterial Patch Antenna Circuitry for Modern Applications", International Journal of Emerging Technology and Advanced Engineering, Volume 12, Issue 12, December 2022, DOI: 10.46338/ijetae1222_05.

**Hayder Almizan** received B.S. degree in electrical engineering from University of Kufa, Najaf, Iraq, in 2008 and Higher Diploma degree in aerospace engineering from the Sapienza University of Rome, Rome, Italy, in 2014. He is currently pursuing his M.Sc. degree in electronics and communications engineering at Mustansiriyah University, Baghdad, Iraq. From 2009 to 2012, he worked at the consultant engineering department of Kufa University. Since 2014, he joined the electronics and communications engineering department in Kufa University. His recent research interests include signal processing, neural network, numerical techniques for electromagnetic engineering of metamaterials and artificial structures.

**Marwah Haleem Jwair** was born in 1986 in Baghdad, Iraq. She earned the bachelor and master degrees in Information Engineering College from AL-Nahrain University, Baghdad, Iraq in 2016 and 2018 respectively. She is pursuing her PhD at AL-Nahrain University since 2021. Her main research interests are in Antennas, Microwave devices, Analog and Digital Electronics.

**Zaid Abedul Hassain** was born in Baghdad (Iraq), in 1979. He received the B.Eng. degree in electrical engineering from University of Mustansiriyah, Baghdad, in 2000, and the master's degree in engineering science in 2002. He is currently a Professor and the Director of the Antenna and Microwave, Department of Electrical Engineering, University of Mustansiriyah.

**Yahiea Alnaiemy** was born in Iraq in 1971, in 1990. He was enrolled at the Higher Institution of Telecommunications and Post before transferring to Al-Mustansiriyah University College of Engineering Electrical Engineering Department in 1994. In July of 1998 he received a Bachelor's a Degree in Electrical Engineering. He continued his graduate studies by joining the Iraqi Commission for Computers and Informatics where he received a Higher Diploma in Information Systems in 2001. He enrolled at Diyala University as an instructor in communication Engineering, electrical power, computer and physics departments. In 2009, he granted a scholarship to complete his master degree in electrical engineering at the University of Arkansas at Little Rock, USA. He got his MSc in Wireless Communications from UALR, USA, in 2012. While completing his graduate degree, his research effort has been in the area of antennas and microwave material characterization. In 2022, He got his Ph.D. in electrical engineering from Budapest University of Technology and Economics (BME), Hungary. His current research areas include UWB antennas, EBG structures, metamaterial, GPS, implantable wireless systems, and nanoscale microwave devices. He is an IEEE Member and reviewer in several international Journals since 2012.

**Lajos Nagy** received the Engineer option Communication and PhD degrees, both from the Budapest University of Technology and economics (BME), Budapest, Hungary, in 1986 and 1995, respectively. He joined the department of Microwave Telecommunications (now Broadband Infocommunications and Electromagnetic Theory) in 1986, where he is currently an associate professor. He has been the head of department of Broadband Infocommunications and Electromagnetic Theory in 2007. He is a lecturer on graduate and postgraduate courses at BME on Antennas and radiowave propagation, Radio system design, Adaptive antenna systems and Computer Programming. His research interests include antenna analysis and computer aided design, electromagnetic theory, radiowave propagation, communication electronics, signal processing and digital antenna array beamforming, topics where he has produced more than 100 different book chapters and peer-reviewed journal and conference papers. Member of Hungarian Telecommunication Association, official Hungarian Member and Hungarian Committee Secretary of URSI, Chair of the IEEE Chapter APComSocEDMTT.

**Taha A. Elwi** received his B.Sc. in Electrical Engineering Department (2003) (Highest Graduation Award), Postgraduate M.Sc. in Laser and Optoelectronics Engineering Department (2005) (Highest Graduation Award) from Nahrain University Baghdad, Iraq. From April 2005 to August 2007, he was working with Huawei Technologies Company, Baghdad, Iraq. On January, 2008, he joined the University of Arkansas at Little Rock and he obtained his Ph.D. in December 2011 from the system engineering and science. His research areas include wearable and implantable antennas for biomedical wireless systems, smart antennas, WiFi deployment, electromagnetic wave scattering by complex objects, design, modeling and testing of metamaterial structures for microwave applications, design and analysis of microstrip antennas for mobile radio systems, precipitation effects on terrestrial and satellite frequency re-use communication systems, effects of the complex media on electromagnetic propagation and GPS. The nano-scale structures in the entire electromagnetic spectrum are a part of his research interest.

# On the Convex Hull of the Achievable Capacity Region of the Two User FDM OMA Downlink

Zoltán Belső and László Pap

***Abstract*—In multiple access channel systems, such as a mobile communication network, it is important to determine how to share the available resources (for example bandwidth and power) among the users. In recent years, one of the promising scheme is Non-Orthogonal Multiple Access (NOMA), where, unlike the traditional Orthogonal Multiple Access, OMA solution, signals for the different users overlap in some domain (power-domain NOMA, code-domain NOMA, etc). In order to evaluate the performance of any NOMA scheme, we need to compare the achievable bit rates of the users (the capacity region) to an OMA case with comparable parameters (for example, same total bandwidth, same total power and same channel conditions, etc). To make this comparison, we first need to know the capacity region for the OMA cases. Many papers make such comparison without detailing the derivation of the capacity region of the OMA case they compare to [1], [2], [3], [4]. In some cases, we have only one free parameter to choose (for example in uplink frequency division multiplexing systems for two users, it is the bandwidth ratio between the users), and the achievable capacity can be directly calculated for both users depending on the single parameter (hence the boundary of the capacity region is trivial). In other cases, such as downlink frequency division multiplexing systems, even for only two users, we have to allocate optimally two resources between the users: the bandwidth and the base station's available power. Hence, it is far from being trivial to determine which combination is better and where the boundary of the capacity region is. In this paper, we provide a derivation for that case.**

***Index Terms*—Orthogonal Multiple Access (OMA); Non-Orthogonal Multiple Access (NOMA); capacity region**

## I. INTRODUCTION

In the mobile communication networks, where a base station is communicating with multiple stations, there are two different cases: uplink and downlink communication, each with somewhat different problems that need to be solved. In the classical orthogonal multiple access (OMA) system, which utilizes frequency division multiple access for communication, there are two resources to share between the users: the bandwidth and the available transmission power.

In the uplink communication case, where the end users attempt to transmit signal to the base station, all the users have their own transmit power limits. Since the communication is orthogonal, the maximum capacity is reached when all the users transmit at maximum transmission power. The only resource that needs to be shared between the users is the available bandwidth. In the case of two users, we can allocate some portion, denoted by $\alpha \in [0, 1]$, of the whole available bandwidth to one of the users and the remainder to the other. This allocation determines the boundary of the achievable capacity region for both users.

In the downlink communication case, where the base station transmits to many users, the power budget of the base station also has to be split between the users, alongside the available bandwidth. So, in the case of two users, we have two parameters: the bandwidth division ratio $\alpha \in [0, 1]$, and the power division ratio $\beta \in [0, 1]$; both can be chosen independently on a two-dimensional capacity plane. Any choice of these two parameters gives us a capacity limit for both users (with a given channel characteristics) that can be represented on a capacity plane, where one axis is the capacity of the first user and the other axis is the capacity of the other user. The convex hull of these points (capacity pairs) gives us the boundary of the achievable capacity region.

In order to examine the shape of the achievable capacity region on the capacity plane we can fix one of these two parameters and let the other changing within its range: this way we get a curve on the capacity plane. We can calculate these curves for any value of the first parameter. This way we get a curve array, and the convex hull of them also gives the same boundary of the achievable capacity region. One can easily see that most of the parameter value pairs are suboptimal. For example, allocating all the power to one user while not allocating all the bandwidth at the same time cannot be optimal: the second user has 0 capacity, and we can increase the first user's capacity by allocating more power to them without any further decreasing the capacity of the second user. So without any complicated derivation we can say that the optimal corner case is that while allocating all the power to one of the users all the bandwidth must be allocated to the same user. Other cases are not trivial. One guess could be to allocate the same portion of power and bandwidth to the users is the optimal allocation, but we will later see that this is only the case when the channel conditions of the two users are the same (it is easy to see that for that case it is optimal, due to symmetry reasons), but in any other cases there are better allocations. Some textbooks (see [5] fig. 15.29, [6] fig. 6.9) gives us the optimal curve, but without derivation or literature reference.

We can get a sense of this capacity region by plotting a school of curves on a capacity plane by choosing a value for one of the parameters (in this case the power division ratio, $\beta \in [0, 1]$) and run the other (in this case the bandwidth ratio, $\alpha \in [0, 1]$) through the whole range. We can see this curves as a kind of iso-power-ratio lines. The convex hull of all possible curves gives us the capacity region. See Figure 1.

Submitted on 12/04/22

Zoltán Belső and László Pap, Department of Networked Systems and Services, Budapest University of Technology and Economics, Budapest, Hungary
E-mail: {belso, pap}@hit.bme.hu

DOI: 10.36244/ICJ.2023.1.2

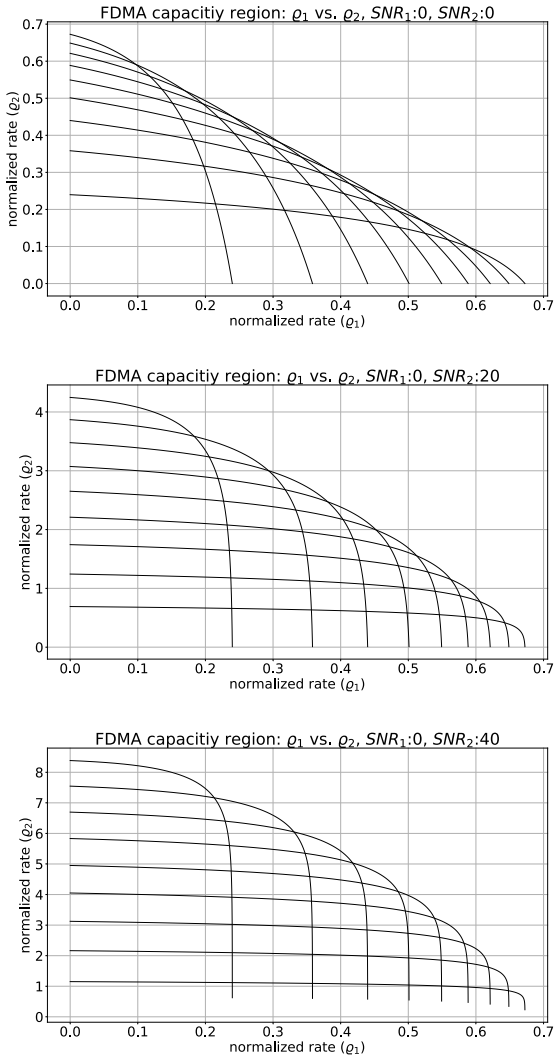



Figure 1. The rate pairs for different bandwidth and power distribution, and the limit of the capacity region

Note, that only one point of each curve falls onto the convex hull, and in the case of differing signal-to-noise ratio the boundary is clearly not a straight line (that would correspond to the case when we chose the same value for both of the resource allocation parameters: $\alpha = \beta$). In some literature (like [4] fig. 4.) the capacity region boundary for the OMA case is graphed as a linear function even in the case of different receiving signal-to-noise ratio (SNR) conditions for the users, which is not correct. There are literature (like [2] fig. 2.) that shows a seemingly correct convex curve as the capacity region of the OMA case, but without derivation or reference. Some others show graphs that are simplified enough not to be conclusive [3].

One can also ask a broader question of what about having more than two users. In this case the situation is even more complicated since we have to split the available resources between all the users. Let's denote the number of users by $N$, then we have to choose $\alpha_1, \alpha_2 \dots \alpha_N \in [0,1]$ and $\beta_1, \beta_2 \dots \beta_N \in [0,1]$ such that $\sum_{i=1}^{N} \alpha_i = 1$ and $\sum_{i=1}^{N} \beta_i = 1$. So we have an $N \times N$ parameter space and find the optimal combination of these parameters to get the convex hull of the achievable capacity region. In this paper, we will concentrate on the two user case.

In this paper, we give a derivation of the curve by expressing the capacity for the second user as a function of the capacity of the first user and one of the two parameters, namely $\alpha$, and finding the extreme value (maximum) of these function with respect of the free parameter $\alpha$, while keeping the capacity limit of the first user fixed. This way, we can determine the maximum capacity that the second user can reach for any given capacity of the first user. We can calculate and plot the limits of the capacity region for different channel conditions of the users.

This paper is organized as follows: First, we provide a formulation of the problem by expressing the normalized capacity of both users with respect of the two free parameters. Then, we express the function for which we need to find the extreme value. After that, we express the limits of the parameters which are applicable for any given capacity of the first user (which we will consider fixed), and examine the function to maximize by giving the graph of it. Next, we express the derivative of the function with respect of the free parameter and find the zero crossing of the derivative. Finally, we plot the found capacity limit pairs of the users. We close the paper with a conclusion section.

## II. Problem formulation

Let's consider the downlink case (OMA) with frequency division multiplexing. In this scenario, we have a base station with a maximal transmitted power limit and an available total bandwidth. Let's consider that we have two users and the base station wants to transmit signal to both users simultaneously.

Here we have two free parameters that can be split arbitrarily between the two users:

- We have to divide the available bandwidth, let's denote the division ratio by $\alpha \in [0,1]$
- We also have to divide the transmitted power of the base station independently. Let's denote the division ratio by $\beta \in [0,1]$

The two users are located in a different physical location, so the channel characteristic may differ for the two users. Let's denote the complex channel characteristic for the two users by $h_i \; i \in (1,2)$.

This way, the achievable bit rate is as follows:[5], [6]

$$\begin{aligned} R_1 &= \alpha W \log_2 \left(1 + \frac{\beta P \left|h_1\right|^2}{\alpha W N_0}\right) \\ R_2 &= (1-\alpha)\, W \log_2 \left(1 + \frac{(1-\beta)\, P \left|h_2\right|^2}{(1-\alpha)\, W N_0}\right) \end{aligned} \tag{1}$$

Where $W$ denotes the total bandwidth available, with $\alpha W$ and $(1-\alpha)W$ representing the allocated bandwidth to user 1 and 2, respectively, $P$ denotes the total transmission power of

the base station, with $\beta P$ and $(1-\beta)P$ are the allocated transmission power to user 1 and 2, respectively. Without loosing generality, we consider that the environmental parameter $N_0$, the noise spectral power density, is the same for both users.

Let's normalize the bandwidth to unit Hertz and get rid of the $\log_2()$ by multiplying with the constant $\ln(2)$. Also divide both the numerator and the denominator of the fraction inside the parentheses representing the signal-to-noise ratio of the users by $W$, so we use the total transmission energy $E = \frac{P}{W}$ instead of the total transmission power $P$:

$$\begin{aligned}\varrho_1 =& R_1 \frac{\ln(2)}{W} \\ =& \alpha \ln\left(1 + \frac{\beta}{\alpha}\frac{E\,|h_1|^2}{N_0}\right) \\ \varrho_2 =& R_2 \frac{\ln(2)}{W} \\ =& (1-\alpha)\ln\left(1 + \frac{(1-\beta)}{(1-\alpha)}\frac{E\,|h_2|^2}{N_0}\right)\end{aligned} \tag{2}$$

Let's further denote the best possible (when all the resources allocated to that user) signal-to-noise ratios for each user as:

$$\begin{aligned}A_1 &= \frac{E\,|h_1|^2}{N_0} \\ A_2 &= \frac{E\,|h_2|^2}{N_0}\end{aligned} \tag{3}$$

Which gives the expressions for $\varrho_1$ and $\varrho_2$ in the following form:

$$\begin{aligned}\varrho_1 =& \alpha \ln\left(1 + \frac{\beta}{\alpha}A_1\right) \\ \varrho_2 =& (1-\alpha)\ln\left(1 + \frac{(1-\beta)}{(1-\alpha)}A_2\right)\end{aligned} \tag{4}$$

The question can be formulated as follows: Given the value of $\varrho_1$, what is the maximal value of $\varrho_2$?

## III. THE FUNCTION TO BE MAXIMIZED

We have two free parameters $\alpha$ and $\beta$, but we don't want to choose them independently, since those would pin down both $\varrho_1$ and $\varrho_2$. Instead, we choose one of the rate (without giving up generality we can choose $\varrho_1$), so we can derive an expression between the parameters, and we can eliminate one of them (again without giving up generality we can choose $\beta$ to eliminate). That way, we can search for the maximum of $\varrho_2$ as a function of $\alpha$ for a given $\varrho_1$.

For a given $\alpha$ and $\varrho_1$ value, we can express $\beta$ as:

$$\exp\left(\frac{\varrho_1}{\alpha}\right) = 1 + \frac{\beta}{\alpha}A_1 \tag{5}$$

$$\beta = \alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right) - 1}{A_1} \tag{6}$$

We got a closed expression for $\beta$ for a given $\alpha$ and $\varrho_1$, which we can substitute into the expression for $\varrho_2$ in (4):

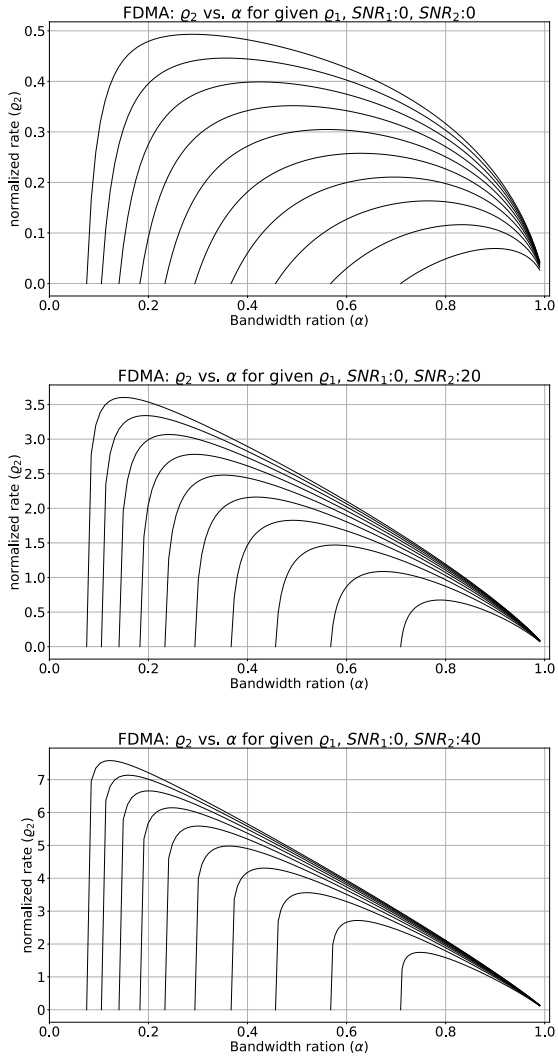



Figure 2. Achievable rate of user2 versus the bandwidth division ratio for a rate of user1. The normalized rate of user1 for the curves from left to right are: $\varrho_1 = 0.10, 0.16, 0.22, 0.33, 0.39, 0.45, 0.51, 0.57, 0.62$

$$\varrho_2(\alpha) = (1-\alpha)\ln\left(1 + \left(1 - \alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right) - 1}{A_1}\right)\frac{A_2}{1-\alpha}\right) \tag{7}$$

This is the function for which we are looking for an extreme value, with respect of $\alpha$. We are looking for such a maximum value for any given $\varrho_1$ value. That way, we get a curve on the two-dimensional $\varrho_1, \varrho_2$ plane.

One can see this optimization problem as follows: There is a range of achievable capacity for one user, that is between 0 and the maximum that can be reached when the total bandwidth and power allocated to that user. Any capacity in between these limits can be achieved by some (possible many) combinations of the free parameters $\alpha$ and $\beta$. The question can be asked is: what is the maximum capacity achievable by the other user for any given rate of the first user?

## IV. EXAMINING THE RATE FUNCTION

In order to graph the rate function in (7) we need to find the limits of the domain $\alpha$.

For the domain of meaningful $\alpha$ values, we can consider the limitations from the original expressions:

Since both $\alpha$ and $\beta$ means a division ratio:

$$0 \leq \quad \alpha \quad \leq 1 \tag{8}$$
$$0 \leq \quad \beta \quad \leq 1 \tag{9}$$

Also, the rates $\varrho_1$ and $\varrho_2$ must be a positive value.

We can expand (9) utilizing the expression for $\beta$ in (6):

$$0 \leq \quad \alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1} \quad \leq 1 \tag{10}$$
$$0 \leq \quad \exp\left(\frac{\varrho_1}{\alpha}\right)-1 \quad \leq \frac{A_1}{\alpha} \tag{11}$$
$$1 \leq \quad \exp\left(\frac{\varrho_1}{\alpha}\right) \quad \leq 1+\frac{A_1}{\alpha} \tag{12}$$
$$0 \leq \quad \frac{\varrho_1}{\alpha} \quad \leq \ln\left(1+\frac{A_1}{\alpha}\right) \tag{13}$$
$$0 \leq \quad \varrho_1 \quad \leq \alpha\ln\left(1+\frac{A_1}{\alpha}\right) \tag{14}$$

This is given as a (greater than 0) lower limit for $\alpha$. The upper limit is $\alpha \leq 1$:

$$\frac{\varrho_1}{\ln\left(1+\frac{A_1}{\alpha}\right)} \leq \quad \alpha \quad \leq 1 \tag{15}$$

As one can see, it is unfortunately a transcendent inequality in terms of $\alpha$, so it doesn't give us a limit for $\alpha$ in a closed form. But we can find the smallest $\alpha$ value, that gives $\varrho_1 \geq 0$ by solving numerically for:

$$\frac{\varrho_1}{\ln\left(1+\frac{A_1}{\alpha}\right)} - \alpha = 0 \tag{16}$$

Solving the limits (16) numerically, we can graph the function from (7) for different $\varrho_1$ values for some SNR cases, see Figure 2. Examining these curves, one can see that there is a unique maximum point for the $\varrho_2$ normalized rate for user2 and a corresponding $\alpha$ parameter value for any given $\varrho_1$ normalized rate of user1.

## V. FINDING THE EXTREME VALUE

Let's find the maximum of the expression in (7) with respect to $\alpha$, that is, calculating $\frac{\partial\varrho_2}{\partial\alpha}$ considering $\varrho_1$ as a constant parameter during the differentiation. We can find the extreme value by solving for $\frac{\partial\varrho_2}{\partial\alpha}=0$ and we can do that for all the possible values of $\varrho_1$.

Since (7) contains a composite function, an $\ln$ with a rather complicated inner expression, we will have to calculate

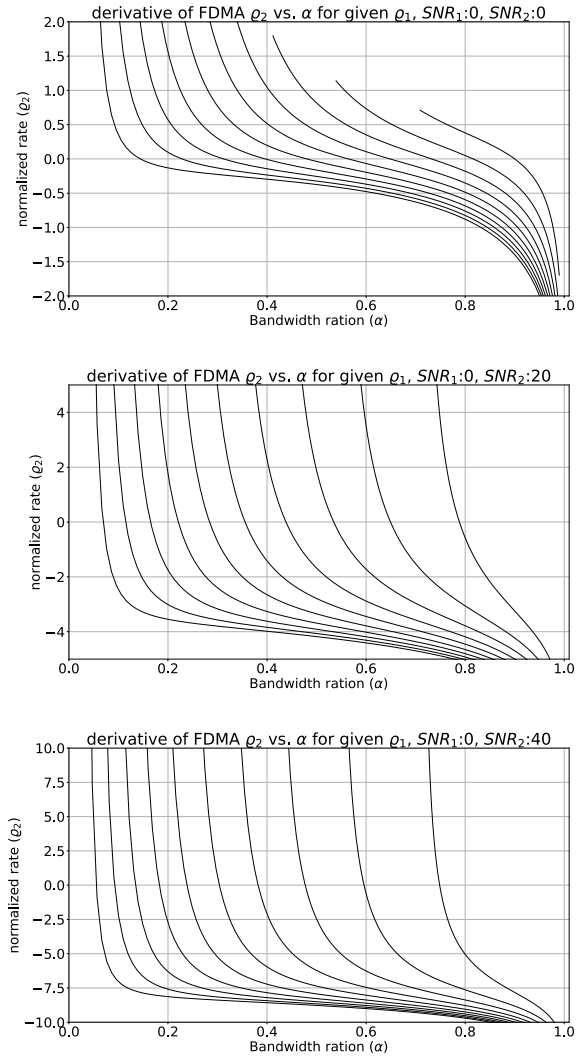



Figure 3. Derivative of achievable rate of user2 versus the bandwidth division ratio for different rates of user1. The normalized rate of user1 for the curve from left to right are: $\varrho_1 = 0.10, 0.16, 0.22, 0.33, 0.39, 0.45, 0.51, 0.57, 0.62$

the derivative of the expression inside the $\ln$ function, let's calculate that first:

$$\begin{aligned}&\frac{\partial}{\partial\alpha}\left(1+\left(1-\alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1}\right)\frac{A_2}{1-\alpha}\right)=\\&=\left(-\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1}+\frac{\alpha}{A_1}\exp\left(\frac{\varrho_1}{\alpha}\right)\frac{\varrho_1}{\alpha^2}\right)\frac{A_2}{1-\alpha}+\\&-\left(1-\alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1}\right)\frac{-A_2}{(1-\alpha)^2}\end{aligned} \tag{17}$$

Using (17) we can utilize the chain rule and write the

derivative of the function in (7) as:

$$\begin{aligned}\frac{\partial \varrho_2}{\partial \alpha} = &- \ln\left(1+\left(1-\alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1}\right)\frac{A_2}{1-\alpha}\right)+\\ &+(1-\alpha)\frac{1}{1+\left(1-\alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1}\right)\frac{A_2}{1-\alpha}}\cdot\\ &\cdot\left[\left(-\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1}+\frac{\alpha}{A_1}\exp\left(\frac{\varrho_1}{\alpha}\right)\frac{\varrho_1}{\alpha^2}\right)\frac{A_2}{1-\alpha}\right.\\ &\left.-\left(1-\alpha\frac{\exp\left(\frac{\varrho_1}{\alpha}\right)-1}{A_1}\right)\frac{-A_2}{(1-\alpha)^2}\right]\\ &=0\end{aligned} \tag{18}$$

This is a transcendental equation, which can only be solved numerically. Figure 3 shows the derivatives for different values of $\varrho_1$.

As can be seen, there is a unique zero crossing of the derivative in the applicable range for all possible values of the $\varrho_1$ parameter, and the derivative goes from a positive value to a negative value. In order to get the boundary of the achievable capacity region we have to solve this extreme value problem for any value of the $\varrho_1$ parameter (the rate for user 1) for the optimal $\alpha$ parameter allocation, and get the value of $\varrho_2$ for that parameter combination, the maximum achievable rate for user 2 at the given conditions.

## VI. THE CAPACITY REGION

Solving (18) numerically for different values for the normalized rate of user1 ($\varrho_1$), we get the maximum achievable normalized rate for user2 ($\varrho_2$).

This pair of rates ($\varrho_1, \varrho_2$ pairs) gives us the boundary of the capacity region on the two-dimensional rate plane. In Figure 4 we plotted this boundary, alongside the school of curves we are getting by fixing the power allocation ratio ($\beta \in [0,1]$) and running the bandwidth allocation ratio ($\alpha \in [0,1]$) the entire range. One can see that these iso-power-ratio curves are touching the boundary only one point each. That point corresponds to the optimal allocation of the other parameter $\alpha$ for the choose value of parameter $\beta$.

In the case of differing signal-to-noise ratios, the boundary is clearly not a straight line. That would correspond to the case when we chose the same value for both of the resource allocation parameters: $\alpha = \beta$. This is the optimal allocation only when both users have the same channel conditions and the same receiving signal-to-noise ratio. The more asymmetrical the channel condition for the users, the more curved the boundary will be. This means that when we want to compare the advantage of a given NOMA communication scheme, we have to compare to this curved boundaries of the OMA scheme.

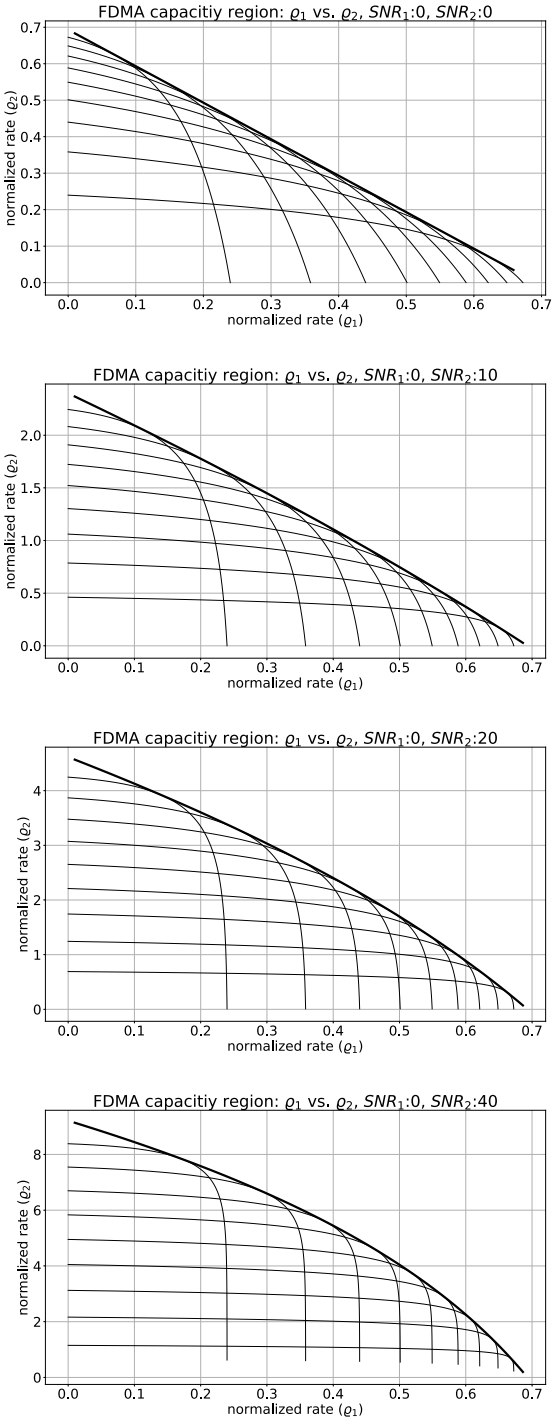



Figure 4. The rate pairs for different bandwidth and power distribution, and the limit of the capacity region

## VII. CONCLUSIONS

We have formulated the problem of finding the achievable capacity region, which is the pair of rates for the two users in the case when the base station transmits signals to both

users (downlink communication) and uses frequency division multiple access scheme. In this case, we must share both the available bandwidth and the available transmission power of the base station between the two users in some ratio, denoted by $\alpha \in [0,1]$ and $\beta \in [0,1]$ respectively in this paper. It is far from trivial to determine which pairs of allocation parameters gives us the best result and what the boundary of the achievable capacity region is.

The first approach idea could be to allocate the two parameters at the same ratio ($\alpha = \beta$), which would give us a linear boundary connecting the extreme points (where all the resources are allocated to one or the other user). However, as we have seen, it would not be optimal when the two users' channel conditions differ. We have to solve an extreme value problem, finding the combination of parameters that gives us the maximum value of the rate for one of user while the other user's rate is fixed.

By careful formulation of the rate functions, we managed to eliminate one of the free parameters by fixing one user's rate as a parameter and find the maximum value of the rate of the other user with respect to the other free parameter. The derivation leads to a transcendental equation, which we solved numerically for different values of the first user rate across the applicable range. This allows us to plot the boundary of the capacity region on the rate pairs plane.

## REFERENCES

[1] Y. Liu, Z. Qin, M. Elkashlan, Z. Ding, A. Nallanathan, and L. Hanzo, "Nonorthogonal Multiple Access for 5G and Beyond," *Proceedings of the IEEE*, vol. 105, no. 12, pp. 2347–2381, 2017, DOI: 10.1109/JPROC.2017.2768666.

[2] K. Higuchi and A. Benjebbour, "Non-orthogonal multiple access (noma) with successive interference cancellation for future radio access," *IEICE Transactions on Communications*, vol. 98, no. 3, pp. 403–414, 2015, DOI: 10.1587/transcom.E98.B.403.

[3] L. Dai, B. Wang, Y. Yuan, S. Han, I. Chih-Lin, and Z. Wang, "Non-orthogonal multiple access for 5g: solutions, challenges, opportunities, and future research trends," *IEEE Communications Magazine*, vol. 53, no. 9, pp. 74–81, 2015, DOI: 10.1109/MCOM.2015.7263349.

[4] L. Dai, B. Wang, Z. Ding, Z. Wang, S. Chen, and L. Hanzo, "A survey of non-orthogonal multiple access for 5g," *IEEE communications surveys & tutorials*, vol. 20, no. 3, pp. 2294–2323, 2018, DOI: 10.1109/COMST.2018.2835558.

[5] T. M. Cover and J. A. Thomas, "Network information theory," *Elements of information theory*, vol. 1, no. 1, pp. 509–612, 2005, DOI: 10.1002/047174882X.ch15.

[6] D. Tse and P. Viswanath, *Fundamentals of wireless communication*. Cambridge university press, 2005, DOI: 10.1017/CBO9780511807213.

**Zoltán Belső** graduates from the Eötvös Loránd University, Faculty of Science as a Computer Scientist (M.S. degree) in 1995. He also graduated as an Electrical Engineer (M.S. degree) from the Technical University of Budapest, Faculty of Electrical Engineering, Branch of Telecommunications in 2007. He is working at the Technical University of Budapest, Faculty of Electrical Engineering, Department of Telecommunications since graduated there as a part-time lecturer. He has worked on Unmanned Aerial Vehicle (UAV) communications systems and Quantum Key Distribution Systems (QKD).

**László Pap** graduated from the Technical University of Budapest, Faculty of Electrical Engineering, Branch of Telecommunications. He became Dr. Univ. and Ph.D. in 1980, and Doctor of Sciences in 1992. In 2001 and 2007 he has been elected as a Correspondent and Full Member of the Hungarian Academy of Sciences. His main fields of the research are the electronic systems, nonlinear circuits, synchronization systems, modulation and coding, spread spectrum systems, CDMA, multiuser detection and mobile communication systems. His main education activity has covered the fields of electronics, modern modulation and coding systems, communication theory, introduction to mobile communication. Professor Pap had been Head of the Dept. of Telecommunications, the Dean of the Faculty of Electrical Engineering at Budapest University of Technology and Economics, and Vice Rector of the University.

# Enhancing Visual Domain Randomization with Real Images for Sim-to-Real Transfer

András Béres and Bálint Gyires-Tóth

***Abstract*—In order to train reinforcement learning algorithms, a significant amount of experience is required, so it is common practice to train them in simulation, even when they are intended to be applied in the real world. To improve robustness, camera-based agents can be trained using visual domain randomization, which involves changing the visual characteristics of the simulator between training episodes in order to improve their resilience to visual changes in their environment.**

**In this work, we propose a method, which includes real-world images alongside visual domain randomization in the reinforcement learning training procedure to further enhance the performance after sim-to-real transfer. We train variational autoencoders using both real and simulated frames, and the representations produced by the encoders are then used to train reinforcement learning agents.**

**The proposed method is evaluated against a variety of baselines, including direct and indirect visual domain randomization, end-to-end reinforcement learning, and supervised and unsupervised state representation learning.**

**By controlling a differential drive vehicle using only camera images, the method is tested in the Duckietown self-driving car environment. We demonstrate through our experimental results that our method improves learnt representation effectiveness and robustness by achieving the best performance of all tested methods.**

***Index Terms*—Artificial intelligence, Neural networks, Reinforcement learning, Self-driving, Sim-to-real transfer**

## I. Introduction

RECENTLY, reinforcement learning-based algorithms have demonstrated strong capabilities in challenging simulated environments, but real-world applications still pose challenges.

Typically, they require large amounts of experience, which can be obtained by training the agents in a simulator. Since simulators are imperfect and incomplete representations of reality, agents' performance typically decreases when transferred back into the real world. This is especially true for agents using cameras, due to the visual differences between the simulated and real environments.

It is advantageous to use cameras as sensors since they are inexpensive, easy to acquire, and can be used for a wide variety of purposes. They record a large amount of high-dimensional data, but algorithmically it is challenging to extract the relevant high-level information from them.

An agent that uses a camera sensor can be trained in a simulator by rendering an image of a simulated camera and using that as input. The difficulty of transferring to the real world stems from the fact that the diversity of images in a simulator is much lower than in the real world. There is a danger that the model learns some specific properties of the simulator (like the colors and textures of some objects), that will not be the same in reality, or will be much more diverse. In that case since these inputs are different from anything the network has seen, its outputs become unpredictable.

In order to increase the robustness of agents to visual changes in their environment, visual domain randomization can be used, which alters the visual characteristics of the simulator in a randomized way between training episodes.

One possibility to further improve performance after sim-to-real transfer would be to include real images in some way into the training procedure. This has the difficulty that while real world images are usually inexpensive to gather, reinforcement learning also requires corresponding rewards, which are more difficult to gather, since they would require precise estimation of the state, which is more error-prone and also more noisy due to imperfect sensors, compared to simulation.

As a result, in order to overcome this limitation while still retaining the ability to utilize real-world data, we propose using an unsupervised method for state representation learning. Our method does not require either rewards or labels in order to learn state representations. Our proposal is to train variational autoencoders (VAE) on both real and simulated frames, so that the training distribution incorporates real frames. Training reinforcement learning agents in simulators using the representations of these pretrained VAE encoders can be transferred and robustly applied to the real world.

## II. Related Work

In order to decrease the gap between the simulation and the real world, a number of techniques can be employed, as follows:

- More realistic simulator environments.
  - More realistic rendering and textures [1].
  - System identification and calibration [2]: more accurate dynamics parameters based on measurements.
  - Novel views of real 3D scenes using Neural Radiance Fields [3][4]
- Domain adaptation [5]: in order to reduce the performance difference between the simulated and real environment, certain statistics can be adjusted to make the

András Béres and Bálint Gyires-Tóth, Department of Networked Systems and Services, Budapest University of Technology and Economics, Budapest, Hungary (E-mail: beres@tmit.bme.hu, toth.b@tmit.bme.hu)

DOI: 10.36244/ICJ.2023.1.3

simulated and real environments more similar, auxiliary loss functions can be applied, or transfer learning can be used.

- Domain randomization [6][7]: A random perturbation of some parameters (e.g. visuals) of the simulated environment is performed in each training episode to broaden the range of environments where the agent performs correctly.
- Regularization:
  - Observation-noise [2]: making the agent more resilient to discrepancies in its observations can increase its robustness.
  - Action-noise [2]: can force the agent to plan more robustly or behave more conservatively.
  - Network regularization [8]: application of techniques typically used against overfitting in deep learning, such as L2 regularization [9], droupout [10], and parameter noise [11].

In the following subsections we introduce those methods, that are most relevant to this work.

### A. Domain Randomization

In domain randomization, selected parameters of the simulator are randomly perturbed during every training episode. By training reinforcement learning agents in a variety of virtual environments, the range of environments they can perform well will be widened, increasing the likelihood of a successful simulation-to-real transfer.

The two main methods of domain randomization are visual domain randomization [6][7], where visual parameters are perturbed, such as textures, lightning, background, and dynamics randomization [12], where the parameters of the process dynamics are changed.

In the case of visual domain randomization and image observations, one could also use image augmentation methods instead of re-rendering the images. These however should not distort the perceived state of the simulator, which is observed by the agent. In first-person view environments such as car driving, random cropping the image would distort the agent's perception of its own position on the track, so we consider it observation-noise instead of visual domain randomization, this however was also shown recently [13][14] to be effective in regularizing the networks to improve training performance. A simple non-distorting image augmentation example is Gaussian pixel noise, but one could change the brightness, contrast or the saturation of the images as well.

These two methods are quite different, and promote generalization in different aspects. While a large diversity can be helpful in the case of visual domain randomization, it is usually detrimental for dynamics randomization. This means that the randomization ranges are important hyperparameters for both methods. In this work we investigate both of them, and will introduce the most relevant works in the following sections.

Visual domain randomization is commonly used for high-dimensional sensors, primarily camera-based tasks, although it may also be applied to LIDARs.

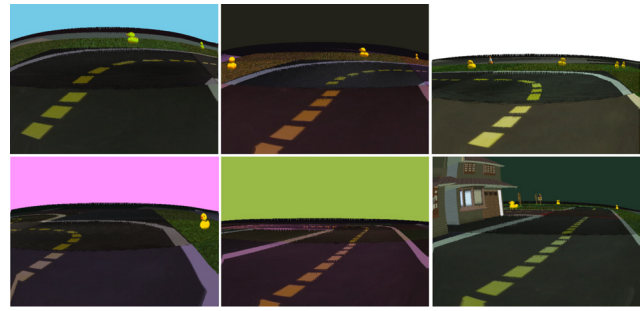


Fig. 1. Example frames with visual domain randomization from the Duckietown Gym self-driving simulated environment.

### B. Direct Visual Domain Randomization

This is the most straightforward way to apply visual domain randomization: the randomized environments are used in the same way as the original one – their observations are directly used for training.

One of the first applications of the technique, in which the goal was to ensure generalization by visual diversity and not to make it visually more realistic, was for the task of indoor camera-based drone control [6]. The authors carried out the training in simulated indoor environments, in which they placed lightsources, furniture, closed and open doors in random positions and directions. They also randomly chose realistic wall textures. Though their network was pretrained on realistic images, they did not use any further real images during training, and their algorithm was capable of flying in the reality as well, with approximately one crash every minute.

The technique was also successfully applied in robotics for object localization [7]. The task of the network was to determine the positions of objects on a table, with other distracting objects present, based on camera images. The training was carried out without any real images, with a random amount of objects with randomized shape, texture and position, and with a random amount of lightsources with randomized direction, temperature and position. They also perturbed the position and direction of the camera, and the parameters of noise added to the images. They used multiple thousand non-realistic textures with randomized colors. Based on the ablation study, the randomized texture and camera positions had the highest impact, which is a finding that we have seen in multiple robotics applications.

The method has also been applied to object detection [15] as well, where the authors used it along with a wide range of image augmentation techniques. Findings show that their model has an accuracy similar to as if it has been trained on a highly realistic virtual dataset. In their case the randomized light sources and textures had the greatest impact on the result.

### C. Indirect Visual Domain Randomization

Indirect approaches do not use the randomized environments for training, but for network regularization or domain adaptation instead.

Invariance regularized domain randomization [16] uses the following idea: the robustness of a policy can be measured by

calculating the average distance between the policy outputs for a randomized observation and the outputs for its original counterpart. So we can add this as a regularization term to the loss function, and use it for optimization, thereby ensuring robustness of the learned strategies.

A similar solution is the following, where the distances are not calculated between the outputs, but between the activations of the last hidden layers instead [17], which can be seen as a high level representation of the input. This helps to avoid the situation where the two parts of the loss function have opposite effects, therefore in this case increasing the strength of the regularization parameter does not cause a performance drop when comparing with [16], as it is shown in the appendix. Another work proposes this same method [18], however an interesting detail is that they use a randomly initialized convolutional layers for data augmentation.

Another method is to train an autoencoder to reconstruct original observations based on visually randomized observations [19], so that the encoder can be used to compress simulator frames to a representation that is invariant to visual changes of the environment.

Visual domain randomization can also be used for domain adaptation [20]. In this case a network is trained to generate a canonical observation (an observation that is similar to the observations of the original environment) based on the randomized observation. This network can also be used to adapt real observations, which we then train our agent on.

A drawback of all works mentioned above is, that they require *paired* canonical - randomized images, which makes incorporating real frames difficult. Real camera images can be considered as randomized observations, but finding their canonical versions is nontrivial and would require image-to-image translation. Another option [21] is to train a generative network to translate simulated images to the domain of real images instead, and use it during simulator-based training.

### D. Sim-to-Real Transfer

It can generally be stated that model-free reinforcement learning algorithms are not using gathered experience efficiently, so they need several interactions with their environment to learn to complete certain tasks. That makes training in the real world slow, and since it also usually needs human supervision, it is generally too expensive and for some applications, such as self-diriving cars, even dangerous to train in the real world. A common solution to this problem is that the agents learn in simulated environments and are then transferred to the real world.

Sim-to-real transfer has already been successfully applied in robot arm manipulation [2], robot locomotion [22] and simple self-driving tasks [23].

However, our simulators can only be imperfect digital twins of reality, so the performance of agents is usually reduced after the transfer.This is called the **sim-to-real gap**, and one has to take it into account if they want to apply agents trained in simulation to the real world.

## III. PROPOSED METHOD

Invariance-regularized methods (see Section II) require paired randomized and canonical images, which would require producing semantically equivalent versions of the real images – which is hard to produce. To overcome this difficulty, we chose to apply the method on top of direct domain randomization methods instead.

Since supervised state representation learning methods require labels (e.g. physical quantities) for each image, they would be noisy to measure and in some cases difficult to obtain in the real world. As an alternative, we used unsupervised representation learning, which requires only real images and no labels, so all what is needed is a camera to take photos.

Based on these considerations, we propose a two stage training method of the reinforcement learning agent.

1) In the first stage, direct visual domain randomization with unsupervised state representation learning is used. The visually randomized simulated images are extended with real images in the training dataset. For unsupervised state representation learning, we utilize variational autoencoders [24][25] with calibrated decoders [26].
2) The trained VAE encoder is then applied to encode the observations to its learned latent space, and the reinforcement learning agent is trained in this latent space. I.e. the simulator output frames are encoded by the pretrained VAE encoder, and these encoded observations are processed by the agent to predict the best possible actions.

The method is depicted on Figure 2.

During state representation learning, the goal is to learn representations that are (1) compact and informative about the environment state, making them useful for the control agent, and are (2) robust to visual changes, making the agent transferable into the real world.

The first criterion can be evaluated by training a reinforcement learning agent using the observations that are preprocessed by the VAE encoder with its weights frozen, and measuring the task performance in the simulator. We test the second criterion by transferring agents into the real world, and evaluating their performance.

#### *Variational Autoencoder*

For state representation learning, we decided to use variational autoencoders as their utility for learning image representations for reinforcement learning algorithms has been demonstrated in previous works [27], including self-driving, both in simulation [28] and the real world [29].

$$\mathbb{E}_{x\in X}[\mathbb{E}_{z\sim q(z|x)}[-\log p(x|z)] + D_{KL}(q(z|x)||p(z))] \quad (1)$$

Equation 1 shows the objective [30] of variational autoencoders, which can be optimized using the loss function in Equation 2, with $X$ being the set of images in the dataset, $x$ an image sample, $p(z)$ being the prior distribution of the latent variables, $q(z|x)$ being the posterior latent distribution an image is encoded into by the encoder, and $p(x|z)$ being the output distribution of pixel values, a reconstruction of the

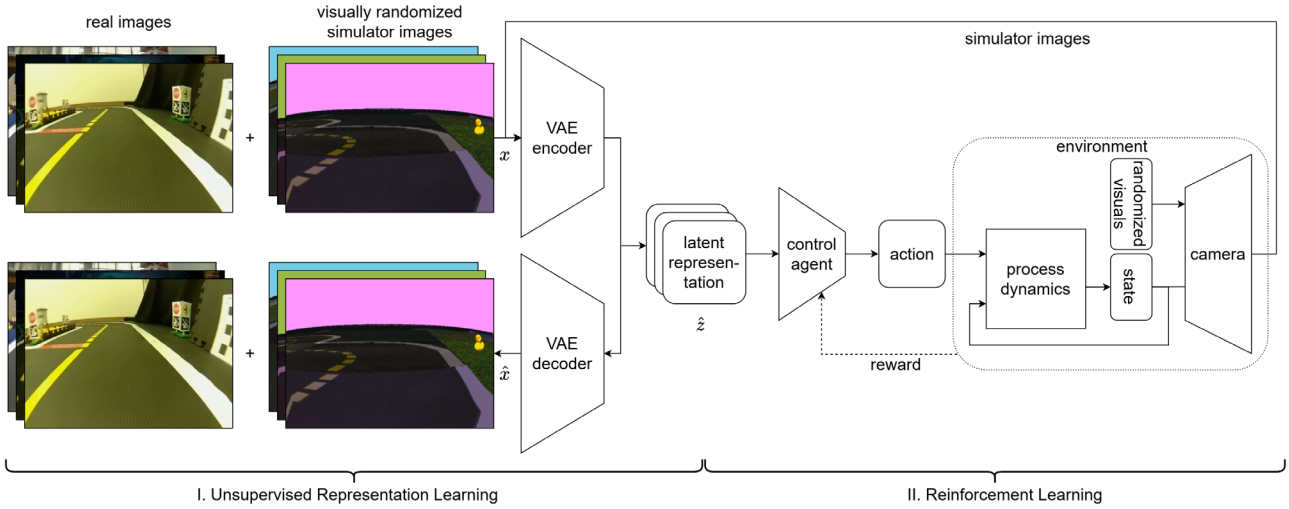



Fig. 2. High level overview of the proposed method. We use real images alongside visually randomized simulator images during unsupervised state representation learning, to improve robustness for sim-to-real transfer. In the first stage we use variational autoencoders for state representation learning, where an encoder is trained to compress the real and simulated images $x$ into latent variables $\hat{z}$ and a decoder is trained to reconstruct them $\hat{x}$. In the second stage a control agent is trained with reinforcement learning and proximal policy optimization in simulator, where the observations are simulated frames preprocessed by the pretrained, frozen encoder network.

original image by the decoder, decoded using a sample $z$ from the posterior latent distribution. In our method $X$ contains both visually randomized simulated and real images.

$$L_{vae} = -\log p(x|z) + D_{KL}(q(z|x)||p(z)) \tag{2}$$

The loss consists of two terms. The negative log-likelihood term measures the reconstruction error between the original images and their reconstructed distributions, and the Kullbach-Leibler divergence term (KL-divergence), which measures the difference between the latent variable distributions the image is encoded into, and their priors.

Sometimes a $\beta$ hyperparameter [31] is introduced, which scales the KL-divergence term to balance the relative strengths of the two loss terms. To eliminate the need of tuning this additional hyperparameter, we used pixelwise calibrated decoder distributions [26], which eliminate the need for it by scaling the reconstruction term instead, by setting the variance of its distribution based on the variance of the pixel values in the training data. For consistency we estimated the variances of the distributions by iterating over the whole training dataset once, and used it throughout the training, instead of estimating it on each minibatch, as was done in [26].

Following previous works, we use a unit Gaussian as the latent prior, and another Gaussian distribution with diagonal covariance matrix as the latent posterior. Using these assumptions the loss terms can be computed analytically, and the reconstruction loss term can be written as Equation 3, and the KL-divergence term as Equation 4.

$$-\log p(x|z) = \frac{(x-\hat{x})^2}{2\sigma_x^2} + \log \sigma_x + \log \sqrt{2\pi} \tag{3}$$

$$D_{KL}(q(z|x)||p(z)) = \frac{\hat{z}^2 + \hat{\sigma_z}^2}{2} - \log \hat{\sigma_z} - \frac{1}{2} \tag{4}$$

The latent posterior mean $\hat{z}$ and standard deviation $\hat{\sigma_z}$ are vectors predicted by the encoder, while the image posterior mean $\hat{x}$ is predicted by the decoder. The image posterior standard deviation, $\sigma_x$, is set beforehand. Removing the constants from the terms, we arrive at our final loss, shown in Equation 5, where $N$ is the minibatch size, $D$ the number of latent dimensions, $X$ the set of images in the batch and $Z$ the set of latent variables.

$$L_{vae} = \frac{1}{2N} \sum_{x \in X} [\frac{(x-\hat{x})^2}{\sigma_x^2} + \frac{1}{D} \sum_{z \in Z} [\hat{z}^2 + \hat{\sigma_z}^2 - \log(\hat{\sigma_z}^2)]] \tag{5}$$

## IV. Evaluation

We implemented and evaluated the proposed method in the Duckietown self-driving car environment [32], on the camera-based lane following task. We used the Stable-Baselines3 [33] reinforcement learning library, and the proximal policy optimization (PPO) reinforcement learning algorithm [34] with a continuous action space.

We evaluated a wide range of baselines not using real images. The examined configurations are listed in Table I, and the baselines are shown in Figure 3, the bottom row is our proposed method. We also included end-to-end reinforcement learning as a baseline, without the application of a pretrained image encoder. Our hyperparameters are detailed in Appendix A, loss functions used for baseline methods are detailed in Appendix B and C.

### A. The Duckietown Platform

The Duckietown self-driving platform consists of multiple main parts, one of which are the Duckiebots, which are small-sized autonomy-capable vehicles, that are controlled by a Raspberry Pi or an Nvidia Jetson Nano, and are equipped with

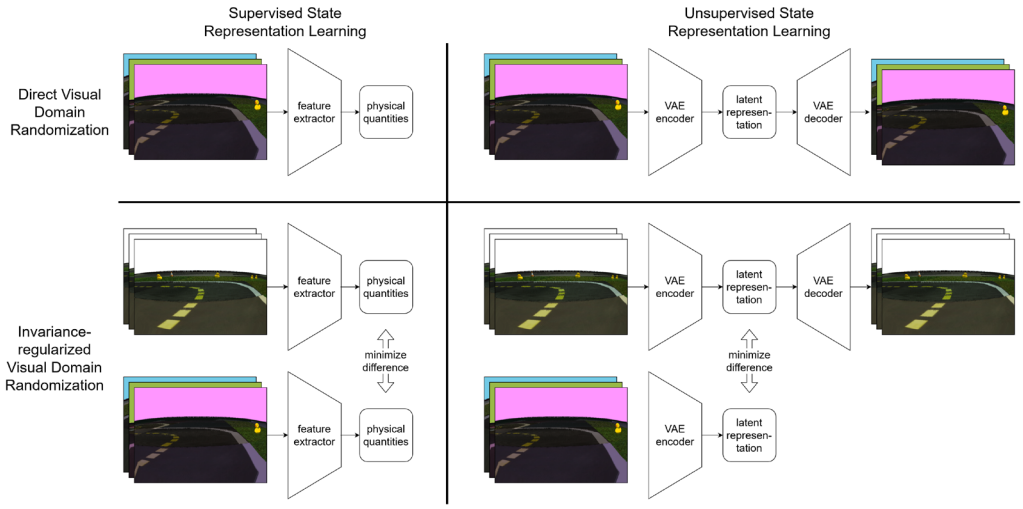



Fig. 3. Benchmarked baseline methods. Direct vs. invariance regularized domain randomization, and supervised vs. self-supervised state representaion learning.

TABLE I
OVERVIEW OF THE BENCHMARKED METHOD CONFIGURATIONS, OUR PROPOSED METHOD IS IN THE BOTTOM ROW IN BOLD.

| Domain Randomization | Encoder Pretraining | Canonical Images | Randomized Images | Real Images |
|---|---|---|---|---|
| None | None | ✓ | ✗ | ✗ |
| None | Supervised | ✓ | ✗ | ✗ |
| None | Unsupervised | ✓ | ✗ | ✗ |
| Invariance-Reg | Supervised | ✓ | ✓ | ✗ |
| Invariance-Reg | Unsupervised | ✓ | ✓ | ✗ |
| Direct | None | ✗ | ✓ | ✗ |
| Direct | Supervised | ✗ | ✓ | ✗ |
| Direct | Unsupervised | ✗ | ✓ | ✗ |
| **Direct** | **Unsupervised** | ✗ | ✓ | ✓ |

a single camera. If a Jetson Nano is present, the camera frames can be processed by it on-device, or alternatively they can be streamed over the network to a remote computer, which is common practice in camera-based robotics [35]. During evaluation we applied the latter option for greater throughput. Duckiebots are differential drive vehicles, they do not use a servo motor for steering, instead their motors are independent on their sides, and they can turn by driving their motors at different speeds.

Another part of the system is Duckietown, which is a small scale well-specified, real, physical driving environment, which can be used by the Duckiebots for driving, therefore their performance can be evaluated in a real environment.

The last main part is the Duckietown Gym, which is a self-driving car simulator, implementing the OpenAI Gym interface. The simulator contains multiple maps that provide tasks such as lane following, navigation in intersections, and pedestrian- (duckie-) and vehicle- (duckiebot-) avoidance. The simulator is capable of simulating multiple agents, opening the possibility for analyzing the joint behaviour of multiple traffic participants [36], however in this work we only consider the single-agent setting.

An important feature of the simulator is that it implements

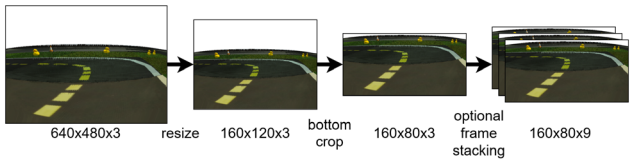



Fig. 4. An illustration of the preprocessing pipeline

visual domain randomization by optionally perturbing the following components:

- Position and color of the light source
- Camera position, angle, and field of view
- Color of the sky
- Texture and color of road tiles
- Amount, type, position and color of environment objects

We perturbed all of these components when training the reinforcement learning agent with visual domain randomization.

### B. Observation Space

Following the work of [37], we downscaled the 640x480 input image by a factor of 4 on both sides to a resolution of 160x120, then we cropped out the upper third of the image, which generally only contained information about the background objects and the sky, which yielded an observation of size 160x80.

Theoretically, stacking multiple past frames can be useful, as this enables the network to infer information about its speed and angular velocity (which need at least 2 frames), and its acceleration and angular acceleration (which need at least 3 frames). These theoretical considerations have been reinforced by prior work [37], and it has also been experimentally shown that stacking more than 3 frames does not yield considerable benefits, therefore we stacked 3 past frames together for every observation for the RL agents when not using rotary encoders. Since we used colored images, the final size of the input image

observations became 160x80x9, as shown in Figure 4. We did not apply any other preprocessing on the input images, such as tresholding certain colors or filtering.

The alternative solution for sensing speed and angular velocity is to use the rotary encoders that the latest edition of Duckiebots are equipped with. This option brings its own set of challenges, and we leave exploring this option to future work.

### C. Action Space

A differential robot is usually controlled by driving its motors on its sides at different speeds. In the case of the Duckiebot, one can control the duty cycles of the pulse width modulated (PWM) signals that drive its DC motors.

That means that the space of possible actions is two-dimensional and by each dimension it spans the range of $[-1.0; 1.0]$. This action space permits some actions that are not useful, for example we do not want the vehicle to drive backwards or to drive it much more slowly than what it is capable of.

Since the task of vehicle control and lane following is inherently continuous, we have chosen to use a continuous action space. We followed prior work [37], and have defined a 1-dimensional action space. The only thing the agent can directly influence is its steering angle, which is then mapped to two target speeds of its two motors. These speeds are chosen to be as high as possible while still having a difference that is proportional to the steering angle. This has the effect that when taking sharp turns the car has to slow down to be able to provide the needed difference between the wheel speeds.

$$u_{avg} = \min(u_{nom}, \frac{1}{1+|\phi|}) \tag{6}$$

$$u_{left} = clip(u_{avg}(1+\phi), -1, 1) \tag{7}$$

$$u_{right} = clip(u_{avg}(1-\phi), -1, 1) \tag{8}$$

The exact derivation is described in Equation 8, where $u_{nom}$ is the desired maximal duty cycle (nominal duty cycle), $u_{avg}$ is the average duty cycle of the two motors (this depends on the desired steering angle), $u_{left}$ and $u_{right}$ are the duty cycles of the corresponding motors, and $\phi$ is the desired steering angle, while $clip(value, min, max)$ is a function that clips its input to be between a minimal and maximal value.

For small values $\phi$ can actually be interpreted as a steering angle in radians, however for larger values it should be interpreted as a scalar value that is proportional to the angular velocity of the vehicle, with $|\phi| = 1$ meaning that either one of the motors stops completely while the other one runs at full speed, meaning that the vehicle goes at half of its maximal speed.

### D. Reward Function

Reward functions for lane following may be based on throttle [28], speed [38], speed parallel to lane [39][40][41][42][29], traveled distance [43], and progress [38], and can include penalties for leaving the lane [28], distance from lane center [42][40][41], or collision [43][41].

Our initial experiments showed that with the default reward function provided by the Duckietown environment, the agents perform suboptimally. Though multiple different reward functions have been tested in previous work [44][45], it has been shown, that a speed-based reward function is already a strong baseline [45]. Based on these results, we have chosen to use a reward function that is physically motivated and is speed-based. In each timestep the reward of the agent is the speed at which it is progressing in its lane (with which speed it is completing the track). A more accurate description is that the reward is the speed of a virtual twin vehicle that moves exactly in the middle of the lane, is exactly parallel to it, and completes its route at the same rate as the actual car.

The exact formula of the reward function is shown in Equation 9, where $v$ is the physical speed of the car, $\delta$ is the signed angle of the car and lane, $r$ is the signed radius of the turn, $c$ is the signed curvature of the turn ($c = 1/r$), and $p$ is the signed distance of the car from the lane.

$$R = v_{progress} = v \cdot cos(\delta) \cdot \frac{r}{r+p} = v \cdot cos(\delta) \cdot \frac{1}{1+cp} \tag{9}$$

The formula can be understood in the following way: $v \cdot cos(\delta)$ is the component of the vehicle's speed that is parallel to the lane, $v \cdot cos(\delta)/(r+p)$ is the angular velocity of the vehicle in a turn, and $v \cdot cos(\delta)/(r+p) \cdot r$ is the circumferential velocity of the equivalent virtual vehicle in the turn, that is moving exactly on the middle of the lane.

This reward function has the advantage that it penalizes high angles and turns taken in the outer regions of the road, while it promotes high speeds and turns taken on the inner regions of the road. In practice the value is generally quite close to simply the speed of the vehicle, which has also been shown to be a plausible reward function [37].

When using this reward function however, care has to be taken to limit how much the car can leave its lane, otherwise it will tend to take left turns by going over to the other lane.

### E. Dataset Collection

To be able to apply state representation learning efficiently, we generated a dataset of observations and corresponding physical parameters. Though one could use a streaming-type dataset, which is generated by the simulator, this would not only bottleneck the training speed, but the samples would not be statistically independent and identically distributed.

Based on these considerations, we generated and saved images of 200.000 scenes during the training of a baseline end-to-end convolutional reinforcement learning agent, and stored 3 different renderings for each image: a visually randomized, a canonical (non-randomized), and a segmented one, as shown in Figure 5. We also saved the corresponding speeds, angular velocities, lane angles, lane distances, and lane curvatures for future work. The generated dataset has a 22.6 GB storage size.

Four different maps were used, all being a part of the official Duckietown simulator (Duckietown Gym [32]): *4way, loop_empty, udem1 and zigzag_dists*.

Note that there are intersections on two out of these four maps, which we included on one hand to increase diversity, but also to help future efforts dealing with intersections.

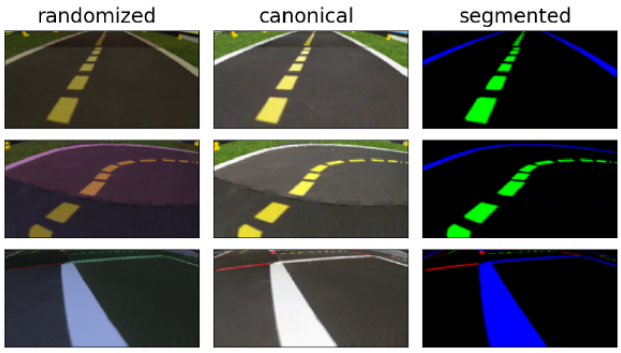



Fig. 5. Samples from the gathered offline dataset showing the 3 different renderings for each scene.

Using the dataset, the mean and variance observations for each image type have been determined on the training set before the training, to enable the usage of calibrated decoders [26].

We have created another dataset as well, which contains 19.000 real images, downloaded from online logs of Duckiebots from the Duckietown website [32]. We handpicked 3 videos of agents with reasonable performance and diverse lighting conditions, extracted and saved the frames from them. These images have been saved under the randomized images category, with no corresponding canonical or segmented frames, nor physical quantities.

### F. Evaluation Metrics

We used the following metrics to monitor the performance of lane following agents during and after the training in simulation:

- Mean angle error: The mean absolute angle between vehicle and the lane
- Mean position error: The mean absolute distance between vehicle and the center of the lane
- Mean completion speed: the same as the mean reward per timestep
- The average length of the evaluation episodes in timesteps, which can be used to calculate the average completion rate, which is the ratio of average evaluation episode length and the maximal episode length

The averages have been calculated over the whole process of the evaluation, which generally consisted of multiple episodes: 10 during training, 50 during final evaluation.

During real world testing, these metrics would be difficult and error-prone to measure, so we used the following evaluation metrics:

- Mean number of traveled tiles: correlates with the distance traveled
- Mean survival time: the ratio of the evaluation time and the number of manual resets required to keep the agent on track

The real world metrics were measured for each algorithm for 60-60 seconds, on both the outer lane and the inner lane of a closed Duckietown track without intersections, with manual resets to the center of the lane if an agent attempted to leave its lane.

Note that the same metrics are used by the Duckietown AI Driving Olympics evaluation [46], but we report mean values instead of medians, which was more practical to measure manually real-time during real world evaluation.

TABLE II
MEAN EVALUATION METRICS OF THE VAE-BASED STATE REPRESENTATION LEARNING METHODS. THE PROPOSED METHOD IS IN THE BOTTOM ROW IN BOLD.

| Algorithm | Reconstr. error [nats/dim] | KL-div. [nats/dim] | Inv. KL-div. [nats/dim] |
|---|---|---|---|
| VAE | 0.125 | 5.5 | - |
| VAE inv. reg. | 0.126 | 3.9 | 8.3 |
| VAE direct | 0.160 | 5.4 | - |
| **VAE direct real** | **0.087** | **4.9** | **-** |

## V. RESULTS

In this section we introduce the results of the first stage, and of the second stage (in simulation and real environments).

### A. Representation Learning (first stage)

Though our main goal was to evaluate the quality and robustness of the learned state representations by training reinforcement learning agents using the pretrained encoders, we also report the pretraining performance metrics of our method and the other baselines.

The metrics of variational-autoencoder based methods are shown on Table II.

Overall the evaluation metrics are in a similar range across the methods, with the exception of the reconstruction error of our proposed method, which is noticeably lower than its counterparts. This could be caused by the fact that both real and simulated images are used for its training, so the distribution of training images is bimodal, while the loss parametrization only assumed a unimodal Gaussian pixel output distribution. This could have led to overestimated output image pixel variance values, making the loss reconstruction error lower under our imperfect assumption.

However our main goal was not to optimize the pretraining metrics, but to evaluate the quality and robustness of the learnt representations by using them for reinforcement learning (in the second stage), so this is what we present in the following sections.

### B. Simulation Environment (second stage)

Table III and IV shows the results of state representation learning algorithms with their default control modules (trained using their representations of their input image type in the Duckietown Gym), evaluated in simulation, evaluated either without or with domain randomization.

Our results in the simulator without domain randomization (Table III) show that from the algorithms that did not use canonical (non-randomized) images for their training (see Table I), our proposed method performs the best. The usage

TABLE III
EVALUATION RESULTS IN THE SIMULATOR WITHOUT VISUAL DOMAIN RANDOMIZATION. COMPLETION RATIOS ABOVE 70% ARE UNDERLINED. THE PROPOSED METHOD IS IN THE BOTTOM ROW IN BOLD. SEE THE CORRESPONDING ROWS OF TABLE I FOR MORE DETAILS ON THE ALGORITHMS.

| Algorithm | Abs. angle error [deg] | Abs. pos. error [cm] | Completion speed [m/s] | Completion ratio [%] |
|---|---|---|---|---|
| E2E | 11.5 | 3.9 | 0.40 | 30.0 |
| SUP | 6.0 | 2.5 | 0.47 | 88.6 |
| VAE | 6.4 | 2.5 | 0.48 | 54.4 |
| SUP inv. reg. | 6.6 | 2.0 | 0.48 | 90.8 |
| VAE inv. reg. | 7.8 | 2.5 | 0.51 | 83.4 |
| E2E direct | 12.9 | 3.2 | 0.35 | 29.3 |
| SUP direct | 9.0 | 3.4 | 0.44 | 39.2 |
| VAE direct | 8.7 | 2.4 | 0.46 | 52.9 |
| **VAE dir. real** | **7.6** | **2.8** | **0.47** | **73.9** |

TABLE IV
EVALUATION RESULTS IN THE SIMULATOR WITH VISUAL DOMAIN RANDOMIZATION. COMPLETION RATIOS ABOVE 70% ARE UNDERLINED. THE PROPOSED METHOD IS IN THE BOTTOM ROW IN BOLD. SEE THE CORRESPONDING ROWS OF TABLE I FOR MORE DETAILS ON THE ALGORITHMS.

| Algorithm | Abs. angle error [deg] | Abs. pos. error [cm] | Completion speed [m/s] | Completion ratio [%] |
|---|---|---|---|---|
| E2E | 10.1 | 3.8 | 0.42 | 23.0 |
| SUP | 17.8 | 4.0 | 0.28 | 12.5 |
| VAE | 11.0 | 3.9 | 0.41 | 18.2 |
| SUP inv. reg. | 6.6 | 2.0 | 0.48 | 84.0 |
| VAE inv. reg. | 7.7 | 2.5 | 0.50 | 82.3 |
| E2E direct | 12.4 | 3.3 | 0.38 | 27.0 |
| SUP direct | 7.3 | 2.8 | 0.47 | 79.9 |
| VAE direct | 6.9 | 2.9 | 0.49 | 82.7 |
| **VAE dir. real** | **6.0** | **2.5** | **0.50** | **88.1** |

of real images might have enabled the network to generalize better to canonical images. However the two invariance-regularized methods do show a higher performance in this evaluation setting. The reason can be that the visual domain randomization in the Duckietown Gym might be too extreme, visual inspection of the images show that the images are usually much darker in comparison to the canonical images. Further evaluation will show however (Table IV, Table V) that they have worse generalization compared to our proposed method in the simulated environment with domain randomization, and also in reality.

End-to-end reinforcement learning seems to have underperformed in these experiments. As one can see on tables III, IV and V, end-to-end reinforcement learning did not manage to perform well in any of the evaluation settings.

Table IV shows our evaluation results in the simulator, with domain randomization. They show that all investigated state representation learning methods that are trained using some sort of visual domain randomization, direct or invariance regularized, are capable of solving the lane following task in the simulator with randomized visuals at near 80% or higher completion ratio. However our proposed method (bottom row) outperformed all of them, achieving the highest, 88% completion ratio.

TABLE V
EVALUATION RESULTS IN REALITY. SURVIVAL TIMES GREATER THAN OR EQUAL TO 20 ARE UNDERLINED. THE PROPOSED METHOD IS IN THE BOTTOM ROW IN BOLD. SEE THE CORRESPONDING ROWS OF TABLE I FOR MORE DETAILS ON THE ALGORITHMS.

| Algorithm | Traveled tiles (outer) | Traveled tiles (inner) | Surv. time (outer) [s] | Surv. time (inner) [s] |
|---|---|---|---|---|
| E2E | 30 | 30 | 5.5 | 4.6 |
| SUP | 0 | 0 | 0.0 | 0.0 |
| VAE | 22 | 26 | 5.0 | 4.6 |
| SUP inv. reg. | 41 | 44 | 60.0 | 30.0 |
| VAE inv. reg. | 43 | 48 | no resets | no resets |
| E2E direct | 30 | 29 | 7.5 | 6.7 |
| SUP direct | 35 | 41 | 10.0 | 20.0 |
| VAE direct | 40 | 49 | 30.0 | no resets |
| **VAE dir. real** | **44** | **51** | **no resets** | **no resets** |

### C. Real Environment (second stage)

Table V shows the results of state representation learning algorithms evaluated in the real world. Our proposed method was able to achieve 60 seconds of driving in both directions without any resets, while also covering more distance than any other method. Since all methods were run at the same speeds (same DC motor pulse-width-modulation duty cycle of 50%), more traveled distance signals a smoother driving with fewer oscillations, which lines up with what we saw visually.

Our experiments also showed that none of the models trained without visual domain randomization (top three rows in Table V) were able to have good performance during real testing. On the other hand, from the six models trained using direct or invariance regularizing visual domain randomization, four were able to achieve an average survival time of 30 seconds or above.

The trends are that supervised state representation learning outperforms end-to-end reinforcement learning, but variational autoencoder-based unsupervised state representations perform the best. In terms of domain randomization, direct methods outperform ones in which it was not used at all, but invariance regularizing methods perform the best, with the exception being our proposed method, which used direct visual domain randomization, but also used real images for training.

We also took part with preliminary versions of the proposed method in the Urban League of the 5th and 6th editions of the AI Driving Olympics [47], and achieved first prize in the Lane Following category in the 5th edition [48], and achieved third and fourth place in the Lane Following with Intersections and with Vehicles categories respectively in the 6th edition [49].

## VI. CONCLUSION

In this work we proposed a novel method for learning effective image representations for reinforcement learning, whose core idea is to train a variational autoencoder using visually randomized images from the simulator, but include images from the real world as well, as if it was just another visually different version of the simulator.

We evaluated the method in the Duckietown self-driving environment on the lane-following task, and our experimental results showed that the image representations of our proposed method improved the performance of the tested reinforcement

learning agents both in simulation and reality. This demonstrates the effectiveness and robustness of the representations learned by the proposed method.

We benchmarked our method against a wide range of baselines, and the proposed method performed among the best in all cases. Our experiments showed that using some type of visual domain randomization is necessary for a successful sim-to-real transfer. Variational autoencoder-based representations tended to outperform supervised representations, and both outperformed representations learned during end-to-end reinforcement learning. Also, for visual domain randomization, when using no real images, invariance regularization-based methods seemed to outperform direct methods.

Based on our results, we conclude that including real images in simulation-based reinforcement learning trainings is able to enhance the real world performance of the agent – when using the two-stage approach, proposed in this paper.

## APPENDIX A
## HYPERPARAMETERS

Tables VI and VII show the hyperparameters used for pretraining and finetuning, while tables VIII and VIII show the used encoder and decoder architectures. Note that the neural network architectures were chosen to be able to fit into the embedded hardware at a limited runtime.

TABLE VI
VARIATIONAL AUTOENCODER HYPERPARAMETERS

| Name | Value |
|---|---|
| Learning rate | 1e-3 |
| Number of epochs | 20 |
| Width | 64 |
| Number of latent dimensions | 8 |
| $\sigma$ parametrization | $\log \sigma$ |

TABLE VII
PPO REINFORCEMENT LEARNING ALGORITHM HYPERPARAMETERS

| Name | Value |
|---|---|
| Optimization steps | 64 * 2048 |
| Learning rate | 3e-4 |
| Number of steps between updates | 2048 |
| Batch size | 64 |
| Optimization epochs | 10 |
| Time horizon (discount factor) | 0.8s (0.96) |
| Gradient clip range | 0.2 |
| Entropy coefficient | 0.0 |
| Initial log standard deviation | -1.2 |

TABLE VIII
ENCODER ARCHITECTURE. ALL CONVOLUTIONS USED A KERNEL SIZE OF 3, STRIDE OF 2, LEFT-RIGHT ZERO PADDING OF 1.

| Layer | Activation | Output dimensions |
|---|---|---|
| Conv | ReLU | width x 40 x 80 |
| Conv | ReLU | width x 20 x 40 |
| Conv | ReLU | width x 10 x 20 |
| Conv | ReLU | width x 5 x 10 |
| Linear | | 2 x latent dim |

## APPENDIX B
## BASELINES WITH SUPERVISED FEATURE EXTRACTION

The loss functions of benchmarked baselines (shown in Figure 3) using supervised image encoders were determined following a similar logic to Section III, to have a fair comparison.

Since the scale of multiple physical outputs that we predict can be very different, we applied a method which works out-of-the box, and doesn't require further hyperparameter tuning. The squared errors of all the physical quantities are normalized by their variance as shown in Equation 10, which is calculated beforehand on the training data.

$$-\log p(y|x) = \frac{(y-\hat{y})^2}{2\sigma^2} + \log\sigma + \log\sqrt{2\pi} \qquad (10)$$

$y$ is the ground truth label (vector of physical quantities to be estimated), $\hat{y}$ is produced by the image encoder, and $\sigma$ is the standard deviation of the output distribution. Removing the constants we arrive at the mean squared error loss, up to a constant scaling factor, shown in Equation 11, with $X$ and $Y$ being the set of images and corresponding labels in a batch.

$$L_{sup} = \mathbb{E}_{x,y\in X,Y}[-\log p(y|x)] = \frac{1}{2N}\sum_{y\in Y}(y-\hat{y})^2 \qquad (11)$$

## APPENDIX C
## BASELINES WITH INVARIANCE REGULARIZATION

For those benchmarked baselines that use invariance regularization-based visual domain randomization (shown in Figure 3), the KL-divergences were calculated between the decoder output distributions that were produced using a canonical and a randomized image.

These KL-divergences are added as auxiliary losses without reweighting to the training loss, which is motivated by a related work [17] (Appendix B), which shows that the network's performance on the reference domain does not depend heavily on the weight of the invariance loss, if it is not the controller's output which is regularized, but an earlier layer. This is true in our case, since reinforcement learning agents (parametrized by multilayer perceptrons) are trained later on top of the image encoders.

For the variational autoencoder, the auxiliary loss is shown in Equation 12.

$$KL(q(z|x_r)||q(z|x)) = \frac{(\hat{z}_r-\hat{z})^2 + \hat{\sigma_r}^2}{2\hat{\sigma}^2} + \log\frac{\hat{\sigma}}{\hat{\sigma_r}} - \frac{1}{2} \qquad (12)$$

TABLE IX
DECODER ARCHITECTURE. ALL CONVOLUTIONS USED A KERNEL SIZE OF 3, STRIDE OF 1, LEFT-RIGHT ZERO PADDING OF 1. ALL UPSAMPLING LAYERS USED NEAREST NEIGHBOR UPSAMPLING WITH A SCALE FACTOR OF 2.

| Layer | Activation | Output dimensions |
|---|---|---|
| Linear | ReLU | width x 5 x 10 |
| Upsampling + Conv | ReLU | width x 10 x 20 |
| Upsampling + Conv | ReLU | width x 20 x 40 |
| Upsampling + Conv | ReLU | width x 40 x 80 |
| Upsampling + Conv | Sigmoid | width x 80 x 160 |

The supervised case is shown in Equation 13. Using that visual domain randomization does not change the underlying physical quantities, and the output variance are predefined, the loss can be further simplified to a scaled mean-squared-error loss, shown in Equation 14.

$$KL(p(y|x_r)||p(y|x)) = \frac{(\hat{y}_r - \hat{y})^2 + \sigma_r^2}{2\sigma^2} + \log\frac{\sigma}{\sigma_r} - \frac{1}{2} \quad (13)$$

$$KL(p(y|x_r)||p(y|x)) = \frac{(\hat{y}_r - \hat{y})^2}{2\sigma^2} \quad (14)$$

In the above auxiliary loss equations $x_r$ is the second input image, in our case the visually randomized one, which we want to be invariant to. $\hat{y}_r$ is produced by the image encoder and $\hat{\sigma_r}$ is the standard deviation for the encoder output distribution based on the randomized image.

With that, we get a similar loss formulation to the work on VAEs with consistency regularization [50], if the regularization strength hyperparameter is set to 1.0, but without the task of reconstructing the invariance-input images.

## ACKNOWLEDGMENT

The research presented in this work has been supported by the PIA Project, a collaboration between Budapest University of Technology and Economics and Continental Hungary Ltd with the goal of supporting students' research in the field of deep learning and autonomous driving.

The work reported in this paper, carried out at BME, has been partly supported by the the European Union project RRF-2.3.1-21-2022-00004 within the framework of the Artificial Intelligence National Laboratory.

## REFERENCES

[1] S. James and E. Johns, "3d simulation for robot arm control with deep q-learning," *arXiv preprint arXiv:1609.03759*, 2016. DOI: 10.48550/arXiv.1609.03759

[2] O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray et al., "Learning dexterous in-hand manipulation," *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020. DOI: 10.1177/0278364919887447

[3] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021. DOI: 10.1145/3503250

[4] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM Transactions on Graphics (ToG)*, vol. 41, no. 4, pp. 1–15, 2022. DOI: 10.1145/3528223.3530127

[5] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, pp. 135–153, 2018. DOI: 10.1016/j.neucom.2018.05.083

[6] F. Sadeghi and S. Levine, "Cad2rl: Real single-image flight without a single real image," in *Proceedings of Robotics: Science and Systems*, 2017. DOI: 10.15607/RSS.2017.XIII.034

[7] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 23–30. DOI: 10.1109/IROS.2017.8202133

[8] Z. Liu, X. Li, B. Kang, and T. Darrell, "Regularization matters in policy optimization-an empirical study on continuous control," in *International Conference on Learning Representations*, 2020. DOI: 10.48550/arXiv.1910.09191

[9] A. Y. Ng, "Feature selection, l1 vs. l2 regularization, and rotational invariance," in *Proceedings of the 21st International Conference on Machine Learning*. Association for Computing Machinery, 2004, p. 78. DOI: 10.1145/1015330.1015435

[10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from over-fitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014. DOI: 10.5555/2627435.2670313

[11] M. Plappert, R. Houthooft, P. Dhariwal, S. Sidor, R. Y. Chen, X. Chen, T. Asfour, P. Abbeel, and M. Andrychowicz, "Parameter space noise for exploration," in *International Conference on Learning Representations*, 2018. DOI: 10.48550/arXiv.1706.01905

[12] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1–8. DOI: 10.1109/ICRA.2018.8460528

[13] D. Yarats, I. Kostrikov, and R. Fergus, "Image augmentation is all you need: Regularizing deep reinforcement learning from pixels," in *International Conference on Learning Representations*, 2021. DOI: 10.48550/arXiv.2004.13649

[14] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas, "Reinforcement learning with augmented data," *Advances in Neural Information Processing Systems*, vol. 33, 2020. DOI: 10.48550/arXiv.2004.14990

[15] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pp. 969–977. DOI: 10.1109/CVPRW.2018.00143

[16] M. Aractingi, C. Dance, J. Perez, and T. Silander, "Improving the generalization of visual navigation policies using invariance regularization," *36th International Conference on Machine Learning, Workshop RL4RealLife*, 2019.

[17] R. B. Slaoui, W. R. Clements, J. N. Foerster, and S. Toth, "Robust domain randomization for reinforcement learning," *arXiv preprint arXiv:1910.10537*, 2019.

[18] K. Lee, K. Lee, J. Shin, and H. Lee, "Network randomization: A simple technique for generalization in deep reinforcement learning," in *8th International Conference on Learning Representations*, 2020. DOI: 10.48550/arXiv.1910.05396

[19] A. Amiranashvili, M. Argus, L. Hermann, W. Burgard, and T. Brox, "Pre-training of deep rl agents for improved learning under do- main randomization," *arXiv preprint arXiv:2104.14386*, 2021. DOI: 10.48550/arXiv.2104.14386

[20] S. James, P. Wohlhart, M. Kalakrishnan, D. Kalashnikov, A. Irpan, J. Ibarz, S. Levine, R. Hadsell, and K. Bousmalis, "Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. DOI: 10.1109/CVPR.2019.01291

[21] M. Tim, M. Szemenyei, and R. Moni, "Simulation to real domain adaptation for lane segmentation," in *2020 23rd International Symposium on Measurement and Control in Robotics (ISMCR)*, 2020, pp. 1–6. DOI: 10.1109/ISMCR51255.2020.9263406

[22] J. Tan, T. Zhang, E. Coumans, A. Iscen, Y. Bai, D. Hafner, S. Bohez, and V. Vanhoucke, "Sim-to-real: Learning agile locomotion for quadruped robots," *arXiv preprint arXiv:1804.10332*, 2018. DOI: 10.15607/RSS.2018.XIV.010

[23] A. Bewley, J. Rigley, Y. Liu, J. Hawke, R. Shen, V.-D. Lam, and A. Kendall, "Learning to drive from simulation without real world labels," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 4818–4824. DOI: 10.1109/ICRA.2019.8793668

[24] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013. DOI: 10.48550/arXiv.1312.6114

[25] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 32, no. 2. PMLR, 2014, pp. 1278–1286. DOI: 10.5555/3044805.3045035

[26] O. Rybkin, K. Daniilidis, and S. Levine, "Simple and effective vae training with calibrated decoders," in *Proceedings of the 38th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 9179–9189. DOI: 10.48550/arXiv.2006.13202

[27] D. Ha and J. Schmidhuber, "World models," in *Advances in Neural Information Processing Systems*, vol. 31, 2018. **DOI**: 10.5281/zenodo.1207631

[28] B. Prakash, M. Horton, N. R. Waytowich, W. D. Hairston, T. Oates, and T. Mohsenin, "On the use of deep autoencoders for efficient embedded reinforcement learning," in *Proceedings of the 2019 on Great Lakes Symposium on VLSI*, 2019, pp. 507–512. **DOI**: 10.1145/3299874.3319493

[29] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah, "Learning to drive in a day," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8248–8254. **DOI**: 10.1109/ICRA.2019.8793742

[30] C. Doersch, "Tutorial on variational autoencoders," *arXiv preprint arXiv:1606.05908*, 2016. **DOI**: 10.48550/arXiv.1606.05908

[31] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "beta-vae: Learning basic visual concepts with a constrained variational framework," *International Conference on Learning Representations*, 2017.

[32] M. Chevalier-Boisvert, F. Golemo, Y. Cao, B. Mehta, and L. Paull, "Duckietown environments for openai gym," https://github.com/duckietown/gym-duckietown, 2018.

[33] A. Raffin, A. Hill, M. Ernestus, A. Gleave, A. Kanervisto, and N. Dormann, "Stable baselines3," https://github.com/DLR-RM/stable-baselines3, 2019.

[34] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017. **DOI**: 10.48550/arXiv.1707.06347

[35] M. Balogh and A. Vidács, "Optimizing camera stream transport in cloud-based industrial robotic systems," *Infocommunications Journal*, vol. XIV, no. 1, pp. 36–42, March 2022. **DOI**: 10.36244/ICJ.2022.1.5

[36] G. Hollósi, C. Lukovszki, M. Bancsics, and G. Magyar, "Traffic swarm behaviour: Machine learning and game theory in behaviour analysis," *Infocommunications Journal*, vol. XIII, no. 4, pp. 19–27, December 2021. **DOI**: 10.36244/ICJ.2021.4.3

[37] A. Kalapos, "Applying transfer learning to autonomous driving task," Master's thesis, Budapest University of Technology and Economics, 2020.

[38] A. Santara, S. Rudra, S. A. Buridi, M. Kaushik, A. Naik, B. Kaul, and B. Ravindran, "MADRaS: Multi agent driving simulator," *Journal of Artificial Intelligence Research*, vol. 70, pp. 1517–1555, apr 2021. **DOI**: 10.1613/jair.1.12531

[39] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Harley, T. P. Lil- licrap, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ser. ICML'16. JMLR.org, 2016, p. 1928–1937. **DOI**: 10.48550/arXiv.1602.01783

[40] E. Perot, M. Jaritz, M. Toromanoff, and R. de Charette, "End-to-end driving in a realistic racing game with deep reinforcement learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017. **DOI**: 10.1109/CVPRW.2017.64

[41] Z. W. Xinlei Pan, Yurong You and C. Lu, "Virtual to real reinforcement learning for autonomous driving," in *Proceedings of the British Machine Vision Conference (BMVC)*, G. B. Tae-Kyun Kim, Stefanos Zafeiriou and K. Mikolajczyk, Eds. BMVA Press, September 2017, pp. 11.1– 11.13. **DOI**: 10.5244/C.31.11. ISBN 1-901725-60-X

[42] M. Jaritz, R. de Charette, M. Toromanoff, E. Perot, and F. Nashashibi, "End-to-end race driving with deep reinforcement learning," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2070–2075, 2018. **DOI**: 10.1109/ICRA.2018.8460934

[43] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16. **DOI**: 10.48550/arXiv.1711.03938

[44] P. Almasi, R. Moni, and B. Gyires-Toth, "Robust reinforcement learning- based autonomous driving agent for simulation and real world," in *2020 International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8. **DOI**: 10.1109/IJCNN48605.2020.9207497

[45] A. Kalapos, C. Gór, R. Moni, and I. Harmati, "Sim-to-realreinforcement learning applied to end-to-end vehicle control," in *23rd International Symposium on Measurement and Control in Robotics (ISMCR)*, 2020, pp. 1–6. **DOI**: 10.1109/ISMCR51255.2020.9263751

[46] Duckietown Foundation. Ai driving olympics lane following metrics. [Online]. Available: https://docs.duckietown.org/daffy/AIDO/out/measuring_performance.html

[47] ——. Ai driving olympics. [Online]. Available: https://driving-olympics.ai/

[48] ——. Ai driving olympics 5: Urban league winners. [Online]. Available: https://www.duckietown.org/archives/66156

[49] ——. Ai driving olympics 6: Urban league. [Online]. Available: https://www.duckietown.org/archives/85031

[50] S. Sinha and A. B. Dieng, "Consistency regularization for variational auto-encoders," in *Advances in Neural Information Processing Systems*, vol. 34, 2021, pp. 12 943–12 954. **DOI**: 10.48550/arXiv.2105.14859

**András Béres** received his BSc and MSc degrees from the Budapest University of Technology and Economics (BME) as an electrical engineer, and currently works as a deep learning engineer at the Continental AI Development Center in Budapest. His interests range from deep learning through robotics to embedded systems.

**Bálint Gyires-Tóth** is an associate professor at BME. He conducts research on fundamental and applied machine learning since 2007. With his leadership, the first Hungarian hidden Markov-model based Text-To-Speech (TTS) system was introduced in 2008. He obtained his PhD degree with summa cum laude in January 2014. Since then, his primary research field is deep learning. His main research interests are sequential data modelling with deep learning, self-supervised learning and deep reinforcement learning. He also participates in applied deep learning projects, including time series modelling, anomaly detection, computer vision and conversational AI. He was involved in various successful research and commercial projects. In 2017 he was certified as NVidia Deep Learning Institute (DLI) Instructor and University Ambassador. His latest AI- related research achievements contribute to the recently launched Artificial Intelligence Systems National Laboratory as a subproject leader.

# Machine Learning Use-Cases in C-ITS Applications

Norman Bereczki and Vilmos Simon

***Abstract*—In recent years, the development of Cooperative Intelligent Transportation Systems (C-ITS) have witnessed significant growth thus improving the smart transportation concept. The ground of the new C-ITS applications are machine learning algorithms. The goal of this paper is to give a structured and comprehensive overview of machine learning use-cases in the field of C-ITS. It reviews recent novel studies and solutions on CITS applications that are based on machine learning algorithms. These works are organised based on their operational area, including self-inspection level, inter-vehicle level and infrastructure level. The primary objective of this paper is to demonstrate the potential of artificial intelligence in enhancing C-ITS applications.**

***Index Terms*—C-ITS; ITS; V2X; smart city; machine learning; deep learning**

## I. Introduction

Transportation has a tremendous impact on everyday life. In the last five years, the number of owned commercial vehicles in Europe has risen by almost 5%, reaching the number of 246 million, putting a strain on the transportation infrastructure, making it highly inefficient [1]. Emissions generated by aviation increased 6.8 times between 1980 and 2015, while the annual distance flown increased by more than 75 times [2]. The number of kilometers traveled on trains increased by 1.5 times between 2000 and 2016 [3]. These data confirm that the use of transport shows a rapidly increasing trend. This leads to overload, unexpected situations, traffic accidents, and makes people spend an extra 27 hours standing in traffic annually [4].

To overcome these issues, a possible solution is designing a new transportation architecture capable of handling the load, however most countries are already too crowded with infrastructure. Intelligent Transportation Systems (ITS) provide potential solutions to increase the safety of commuters and optimize traffic flow, thus reducing traffic congestion and air pollution and increasing the reliability of public transportation. If the listed problems are at least partially solved in a region that has an impact on the quality of life [5].

The development of ITS are supported by industrial and academic resources as well, leading to a significant momentum in the field and making it one of the most researched topics. In recent years, Artificial Intelligence (AI) and sensor technology have undergone great development, which has resulted in an increase in AI-based C-ITS applications.

AI is a powerful tool for processing extensive amounts of gathered data to gain insights into transportation systems. The first use of the terminology "AI" goes back to Dartmouth College in 1956 [6]. They held AI is the process that gives the ability for computers to learn by modelling the learning aspects of the human brain. AI-based applications are a popular and effective way for data-driven applications and used for prediction, anomaly detection, etc.

The goal of this paper is to provide a comprehensive study of novel AI-based applications used in C-ITS. Section III introduces the categorisation of C-ITS applications based on the level of cooperativeness and provides a summary about Machine Learning (ML) using C-ITS related examples. Section IV collects machine learning based applications used in C-ITS and organizes them based on their operational area. Then Section V presents research challenges and unsolved issues related to AI-based C-ITS applications. This review collects novel publications of the field and contributes to our understanding of the impact of machine learning in C-ITS. Although there are survey and review papers on machine learning use-cases in smart transportation, however our aim is to organise these applications based on their operational area, showing how AI contributes in every layer of smart transportation. This novel organisation can help the reader understand how each layer plays an important role in the smart city concept.

We tried to collect articles that were no older than 6 years and had relatively high number of citations. We preferred articles published in high-ranked journals and conferences in the filed of transportation, V2X, and computer science. We aimed at ensuring that the selected articles are recent, well-cited and represent the current state-of-the-art of this field. Using this approach we tried to provide an up-to-date survey of machine learning use-cases in C-ITS applications.

## II. Machine Learning

In many scientific fields, such as self-driving cars, vast amounts of data are being generated. This wealth of data enables data-driven approaches that perform well with artificial intelligence-based algorithms.

In recent years, AI and V2X have been extensively studied. The number of AI related publications has increased by over 200% in the last 10 years. The number of filed patents has increased 30 times in the last 7 years, but only the 0.06% of the posted AI related jobs in the U.S. was related to autonomous driving in the last 2 years [7].

AI is often defined as the technique that gives the ability for computers to mimic the human thinking. Machine Learning (ML) is a subcategory of AI. The main essence of ML is not only to let machines use logic but also to give them the ability

The authors are with the Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Budapest, Hungary
(E-mail: {bnorman, svilmos}@hit.bme.hu)

DOI: 10.36244/ICJ.2023.1.4

to learn from experience, thereby improving themselves for a better performance. This is typically done by minimizing a loss function defined as a function of input parameters. Learning is a generic process of modifying an existing knowledge base to better adapt to a situation.

The term ”machine learning” covers a wide range of algorithms and has became a popular word in today's computer science. Most data-driven problems require the ability to recognize complex and abstract connections in the data, which machine learning is highly successful at. This section provides a comprehensive summary of machine learning use-cases and highlights how AI can contribute to various problem categories.

### *A. Generic Machine Learning Workflow*

There is a generic workflow for implementing machine learning models. This section provides a brief summary of these steps and shows examples of how they can be interpreted for C-ITS applications.

1) **Data collecting**: The success of a model highly depends on the amount and quality of the data it is trained on. Althnian et al. [8] and Prusa et al. [9] published a paper which leads to the conclusion that the accuracy of a machine learning model highly depends on the size of the training dataset. Thus, data is highly valuable for fields such as social advertising, recommendation systems, fraud detection.
   *C-ITS application can rely on several data sources. Vehicles can share their state, such as speed and position. An RSU (Road Side Unit) can provide information about the infrastructure. Several data collecting units use computer vision, which is a field of AI, that can derive visual information.*
2) **Feature selection**: Collected data may contain features that contribute little or none to the target variable one would like to predict / classify on. The goal is to find the important features that are highly relevant and remove the features that are not. This process improves the accuracy of the model, reduces computation time, and prevents the model from overfitting. In a comprehensive literature review of feature selection methods, Chandrashekar and Sahin present different algorithms that might be employed for this task [10].
   *High accuracy predictions and proper incident detection, such as an accident caused traffic congestion, are not possible without identifying the correct features. For example, the breaking intensity of a vehicle changes radically when an accident occurs in front of it, but inner temperature is not affected by it. It is essential to build the model on the right features to achieve good performance.*
3) **Initialising the algorithm**: Each and every machine learning algorithm solves different problems. At implementation, attention must be paid for the amount of data, data labeling, cleaning, gap-filling, and other preprocessing tasks. Chosen hyperparameters of an algorithm have a great impact on its performance, so optimisation plays a key role here. As Feurer et al. state in their paper about hyperparameter optimisation, it is a fundamental part of the accurate result, or as the Conclusion states: “The devil is in the details” [11].
   *Goals of our applications can differ. The algorithm must be selected according to its goal, and the amount and quality of data gathered. A bus delay prediction system may be based on time-series analysis and prediction, but a vehicle identifier algorithm require completely different machine learning algorithms.*
4) **Training of the algorithm**: The model needs to be trained on the specific training data from which it can learn. During the training process, the goal is to minimize the problem specific loss function iteratively to improve performance. Just a subset of the collected data is used during the training process. It is important to split the data into train-test-validation sets.
   *A clustering algorithm can also be applied on several datasets. It can be utilised, for example, to detect dangerous drivers, but also to detect traffic congestion events.*
5) **Evaluation of the model**: After the learning process, the model's performance is evaluated on a test dataset that differs from the training set. Performance of the model is often measured by computing various performance metrics, such as accuracy (ratio of correct predictions to the number of all the predictions) , precision (ratio of correctly identified cases to the number of all the identified cases), recall (ratio of correctly identified cases to the number of actual positive cases), F1-score (harmonic mean of precision and recall), ROC (curve of true positive rate against fals positive rate at every classification thresholds), AUC (area under the ROC curve). Hossin et al. give a summary about commonly used metrics [12].
   *Evaluation of newly adapted C-ITS applications is often enhanced by professionals who monitor the traffic and validate the results of the model compared to the real traffic characteristics.*
6) **Implementing feedback from results**: The complex analysis of various accuracy metrics can provide information about the type of errors that the algorithm has made. (Whether it results, for example, high recall but low precision.) This information can tune the model or use dataset balancing metrics.
   *Before a new application gets implemented in live transport system, it is preceded by a great amount of simulation to test every possible scenario. A new, adaptive traffic light controlling system, for example, might be dangerous to implement before they fully test it because it can cause serious consequences, such as accidents.*

The target variable, or label is a discrete or continuous value that is assigned to an entity. Machine learning algorithms can be categorised based on the occurrence of the target variable. The following sections provide details, use-cases and common algorithms for the four main categories:

### B. Supervised Learning

Supervised learning is a type of ML where a target variable is assigned to every entity in the dataset. The main goal is to approximate the value of the label based on other features. There are two main types of problems in supervised learning:

**Regression** is a statistical approach. Regression uses a function created from the combination of features in order to estimate the value of the continuous target variable. Regression can be used to model the relationship between different variables that affect the transportation. Regression based models can be used to predict the influence of different impacts on the transportation, such as weather conditions to traffic speed or travel time.

**Classification** is the process of arranging data points into pre-defined groups based on their features. The target variable is discrete and represents a group. Based on the number of classes, two categories are distinguished: OCC (One-class classification) which is used to predict whether the data is in one specified class and the Multiclass classification (MCC) which classifies the data into 3 or more classes. MCC is often solved with multiple applications of OCC. When labeled data is available, anomaly detection problems can be approached as an unbalanced classification problem. Commonly used algorithms include various types of decision trees, multi-layer perceptron, regression trees, logistic regression, support vector machine, etc. [13].

### C. Unsupervised Learning

The primary difference between supervised and unsupervised learning is the absence of the target variable. In unsupervised learning, there is no explicitly given label on the data that can be used as the target of the prediction. In most cases, the goal is to find subsets with similar attributes and perform an action on them.

**Clustering** based algorithms find structure in the dataset, thus creating groups in the data containing similar data points. Hyperparameters of clustering algorithms include the number of clusters, similarity metric, etc. Compared to supervised learning problems, the miss of the target variable indicates that there are no optimal values for these hyperparameters. The result given by a clustering algorithm highly depends on these hyperparameters, but the best result is not easily defined because there is no target variable we can measure the accuracy on. Clustering algorithms can be hierarchical or partitional. Hierarchical clustering results clusters within clusters, while there is no hierarchical relationship between the clusters at partitional clustering. Common algorithms are k-Means, DBSCAN [14].

**Anomaly Detection**

The goal of anomaly detection is to find data points that do not conform to the dataset's expected behaviour. Anomaly (or as it is often called, outlier) detection is often highly specific to a given domain. Anomaly detection is approached in several ways: as classification problem, clustering problem, nearest-neighbour search, evaluation of different statistics or information theoretic and spectral metrics[15]. Anomaly detection can help to filter noise or identify malicious activities. Commonly used ML technologies are AutoEncoder Neural Networks, Isolation Forest, etc [16].

**Dimension reduction** is a process that reduces data from a high dimensional space to a lower dimensional latent space using mathematical operations (e.g., different projections) with minimal loss of information. It can be beneficial in several aspects such as reducing the size of the data, improving performance, allowing high-dimensional data to be visualized, and reducing overfitting.

PCA is one of the most popular early statistical technique that is used for dimension reduction. It projects the data into a smaller subspace with the methods of linear algebra. Other popular algorithms are t-SNE, UMAP [17].

### D. Deep Learning

Deep learning is a machine learning technique learning representation by examples. It is a technique that is used to try and learn like human beings by modelling the architecture of brain cells.

Neural networks are computational models inspired by the structure and function of the biological neurons in the brain. They consist of interconnected nodes (neurons) that can process and transmit information, allowing it to learn patterns and relationships in data. Using neural networks instead of classical machine learning methods can lead to deeper recognition capability in higher dimensions. These networks mimic the functioning of biological neurons in the brain. A neuron receives an input signal, performs computation on it, then passes the output signal to the next layer. The output of the neuron is the weighted summary of its inputs, passed through an activation function. The main goal of using activation function is to introduce non-linearity into the output, thus allowing the neural network to learn complex, non-linear patterns as well. There are several activation functions, each has its own advantage and the choice depends on the problem's requirements. Commonly used activation functions are [18]:

- *Sigmoid:* The goal of sigmoid function is to map the input into the interval between 0 and 1.
- *ReLU:* Rectified Linear Unit function sets values that are less than 0 to 0 and leaves others unchanged.
- *LeakyReLU:* Similar to ReLU, but for negative inputs, instead of zero it introduces a small slope.
- *ELU:* Similar to LeakyReLU, but it smoothens the curve for negative inputs, thereby improves learning speed and prevents overfitting.
- *Hyperbolic Tangent:* It maps the input to a range between -1 and 1.
- *Swish:* Swish is a relatively new activation function. It is similar to ReLU, but it has a learning parameter and in some cases it can outperform ReLU.

### E. Other categories

*1) Semi-Supervised Learning:* Labeling data is often difficult, and labels are not always available for the dataset. Semi-supervised learning combines both supervised and unsupervised learning by using large amounts of unlabeled data, together with a significantly smaller amount of labeled data.

This approach is often used when trying to predict cases for which there are no examples in the training set, thus is a good tool for anomaly detection, malicious activity detection and disaster forecasting.

*2) Reinforcement Learning:* The goal of reinforcement learning is to learn a specific behavior by taking actions and receiving feedback in the form of positive or negative reinforcement. The model learns complex action sequences by trying different combinations and improving itself based on the received feedback. One of the most common type of reinforcement learning is Q-learning, where a table or a function is used to determine the optimal action in a given state.

## III. COOPERATIVE INTELLIGENT TRANSPORTATION SYSTEMS (C-ITS)

A big breakthrough in the development of ITS has occurred when new communication capabilities of vehicles and the infrastructure became possible. V2X (Vehicle-to-X or Vehicle-to-everything) is a generic term referring to the usage of communication capabilities of vehicles and transportation infrastructure elements, based on standardized architectures and protocols. V2X communication technology enables vehicles, roads, and pedestrians to link up and exchange information about their environment and state. This communication results in improved transportation efficiency and safety. Through V2X, vehicles can share their measures with other participants, facilitating new cooperative behaviors. V2X based applications can improve various aspects of transportation. Its main goal is to make transportation more efficient and safer. The elements of transportation can have static (number of lanes, structure of intersections, speed-limit, etc.) and dynamic information (position of vehicles, traffic load, etc.). In C-ITS, information exchange through V2X enables static and dynamic information to be shared among ITS elements. The sharing of information from other ITS also allows for cooperative behavior to improve ITS to C-ITS. A common approach is to use cloud-based services that enables vehicles to exchange information. In their work, Chen et al. [19] propose a cloud-based traffic control system. They provide a platform to support cross-sector information sharing. Their system can provide vehicle's weather data, can alert them about traffic congestions and emergency vehicles. Vieria et al. [20] presents a roadside and cloud architecture. Their system has been deployed and is used on different motorways in Portugal. The system also offers traffic visualization and event reporting. The system is not dependent of the underlying communication technologies, thereby it is easy to implement.

*1) Radio Access Technologies:* C-ITS applications have strong QoS requirements. With the use of V2X, the development of reliable, low-latency radio access technologies (RATs) also became important. Nowadays two RAT competes each other: **802.11p** for dedicated short-range communication (DSCR) and **New Radio 5G V2X (NR V2X)** for cellular V2X (C-V2X). 802.11p is an IEEE 802.11 standard, also known as Wireless Access in Vehicular Environments. It was introduced specifically for V2X communication. It operates in 5.9 GHz and provides low-latency, reliable communication. The Next-Generation Wireless Access for Vehicular Environments (802.11bd) protocol is a more recent standard that builds on 802.11p. It is more flexible, has improved performance and uses both 5.9 GHz and 60 GhZ bands. 5G is the 5th generation of wireless communication technology. 5G can transmit large amount of data quickly, thus enabling vehicle to share a vast amount of information with low latency. 5G network is also highly reliable and has built-in security features that are also important. Both DSCR and C-V2X are promising technologies that can provide reliable, low end-to-end latency communication. C-V2X has the potential to bring additional benefits, such as better coverage for V2X, reducing infrastructure deployment costs and increased deployment flexibility. Nevertheless, 802.11p technology is ready to be used, and several applications have been developed using it, as this standard has been on the market for longer. Europe and the United States, the 802.11p standard is more widely used, while China's V2X developments are based on the NR-V2X RAT [21] [22].

*2) GeoNetworking protocol [23]:* GeoNetworking is a protocol in the network layer which makes location-based message forwarding possible. It transports messages over IEEE 802.11p in GeoNetworking packets. It provides services for top layer protocols. There are 3 routing ways that are shown in Fig. 1 below. The GeoUnicast messages are sent from a node to one receiver. The GeoBroadcast message can be concentrated in a physical area (rectangle, square) and the topologically scoped broadcast messages are sent in the environment of the sender node. It allows messages to be received only by nodes for which the information is useful.

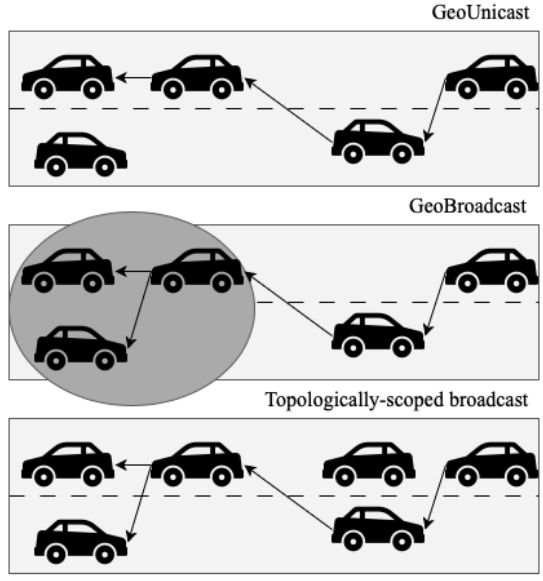



Fig. 1. Package forwarding in GeoNetworking

*3) Facility layer protocols:* Entities on or near the road can communicate in a wide range of situations. A roadside unit, a traffic light, or a car that travels 130 km/h has to communicate different information to other participants thus

several protocols are utilized use in V2X. Communication using unified protocols is important, so standardization is a key aspect of successful communication. This section highlights a few of the most common protocols that C-ITS applications use.

- **CAM** - Cooperative Awareness Message [24]:
  Contains state information (velocity, vehicle attributes, etc.) about the represented entity. Messages sent with CAM are containerized. They are sent in 1-10Hz periods, providing up-to-date information. (The proper sending frequency plays an important role in efficiently using the communication channel. A vehicle travelling at 130 km/h will send a message every 3 meters, if the sending frequency is 10 Hz. In a traffic jam, the vehicle may not even move 1 meter in a second and using 10 Hz sending period in this scenario would be unnecessary.)
- **DENM** - Decentralized Environmental Notification Message [25]:
  Used for sharing incident information (accidents, road construction, etc.) on the road. It can provide information about incidents with constant or static positions. DENM messages can be sent from a vehicle by OBUs (On Board Unit) and from the infrastructure by RSUs (Road Side Unit). DENM contains information about the incident, such as its starting time, its estimated ending time, and describes the triggering event.
- **CPM** - Cooperative Perception Message [26]:
  Allows vehicles to share observations of their local environment based on their sensor data. The messages are sent with 1-10Hz frequency, similar to CAM. CPM often carries the data used for self-driving systems. CPM is a novel development.
- **MAPEM, SPaTEM** - MAP Extended Message [27]:
  Messages from the infrastructure to the ITSs. The MAPEM and SPaTEM describe the lanes and the sequences at crossroads.
- **SREM** - Signal Request Extended Message [28]:
  The vehicle can request modifications in the traffic lights sequence at a crossroad. The request is prioritized on a scale from 1 to 14. It is especially useful for ambulance cars or fire trucks so they can request priority in the traffic.
- **VAM** - Vulnerable Road User Awareness Message [29]:
  Used for describing the position of participants that are not vehicles, such as pedestrians or cyclists. The vehicles can be alarmed about them, thus avoiding unexpected situations and preventing accidents.

### *A. Dynamic messages in C-ITS*

Transportation can have static and dynamic properties. Static information is, for example, the degree of a curve or a position of a crossroad. These properties are permanent and do not change in time. The constant change in the traffic situations generates dynamic data as well. The position of a vehicle, the occupancy of the infrastructure, average speed, velocity, etc. are always changing. Based on these kinds of dynamically changing data, we can classify the evolution of V2X applications into the following categories:

*1) Day 1 - Share sensed information:* ***Driving Awareness****:* Shared information is valid only in the local environment of the sender. Vehicles can be aware of their environment, and the most important parameters of those travelling nearby. The vehicle receives and processes messages coming from other vehicles in its communication range. (The size of a vehicle's range is changing based on conditions such as travelling speed, infrastructural grid.). Day 1 applications are simple and do not interfere directly with driving. A good example is Cooperative Adaptive Cruise Control (C-ACC). The vehicle accelerates and slows down based on the received messages. (It's important to understand that the basic ACC is performed by the vehicle's local sensors, while the C-ACC relies on data provided by other vehicles.) RWW (Road Works Warning) can inform the driver about ongoing road maintenance. Collision avoidance works by the alert coming from another vehicle.

*Day 1 applications are based on the communication capabilities of the participants. It relies on them being capable of sharing their status, thus alerting one another.*

*2) Day 1.5 - Share sensed information extended to multi-modal applications:* This category includes multi-modal applications that are based on shared sensing. These applications rely on shared data and improve mostly the utilization of the infrastructure. These applications can be divided into 6 main groups:

- Parking: Provide information about free parking spots, thus minimizing the unnecessary traffic and environmental load.
- Intelligent Routing: Information providing services which can help electric cars to find the optimal charger regarding to the traffic and the destination. These services can implement adaptive routing.
- Transporting: Help manage and support the transport vehicles in cities.
- Safety: Improves the safety of pedestrians and cyclists. Can function as a recommendation system for entities with limited visibility.
- Collision: Can reduce the possibility of a collision in critical situations, such as overtaking or in a turn with a bad viewing angle. For example, vehicles are able to assist in the decision making whether or not to perform an overtake.
- Direction Alert: Can alert the driver and its environment if a vehicle is going against the good driving direction.

*3) Day 2 - Sharing sensor measurements:* ***Cooperative Sensing****:* These applications extend the use-cases of Day 1 applications. The main difference compared to Day 1 applications is besides sharing measurements and information about the entity, they also share sensor measurements about their environment. These applications share their sensed environment, thus making other entities able to create a more comprehensive model of their own environment.

The CPM (Collective Perception Message) application proposes a standardized method to share the sensed objects in the vehicle's environment and the free space between these objects. The gathered data can be used as an input for more complicated, sensor-fusion based applications, further improving V2X applications.

*4) Day 3, 3+ - Cooperative and synchronized cooperative driving:* ***Coordinated maneuvering****:* Maneuvering Coordination Service (MCS) is a Day 3 service, and it lets vehicles share their intentions with others [30]. These applications implement a higher level of self-driving. A fully automated transportation can reduce the number of accidents to 0 and can reach the optimal use of infrastructure both in the aspect of travelling time and pollution.

### *B. Hierarchical layers of C-ITS Applications*

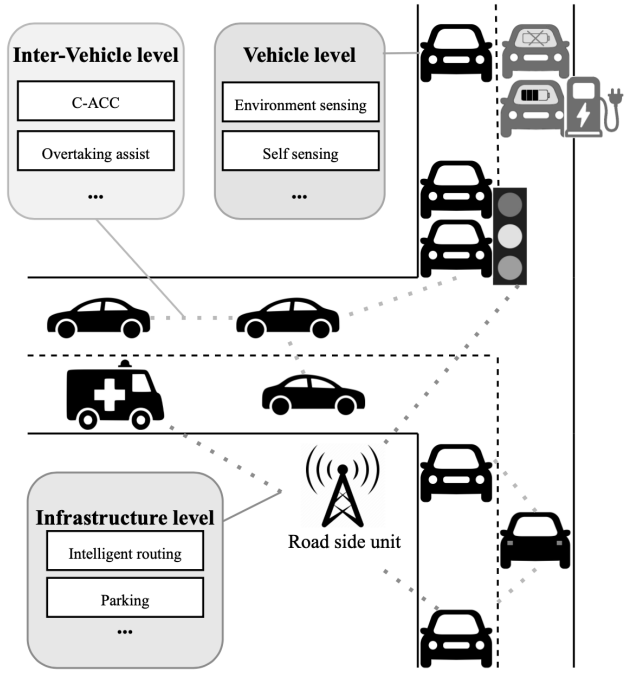



Fig. 2. A model of a smart transportation infrastructure

Fig. 2 shows a model of the smart transportation concept. This paper distinguishes 3 levels of C-ITS applications that are listed in this section. Following sections provide detailed information on:

- **Vehicle level** applications operate on the level of vehicle. They can sense self-states, such as faults, driver tiredness, and the environment of the vehicle, such as object recognition, collision detection.
- **Inter-Vehicle level** applications operate in a group of close vehicles. These applications operate with shared data and can not be realized by a single vehicle.
- **Infrastructure level** applications impact the whole transportation infrastructure.

In order to ensure the success of smart transportation, it is crucial for the different layers to heavily rely on each other. The flowchart of Fig. 3 is an example of a C-ITS application that uses all abstraction layers demonstrated in this paper.

When a vehicle recognizes an ambulance vehicle, but it is unable directly notify every other vehicle, or an infrastructural element (it may be out of the range of any RSU). Therefore, it sends an alert to the vehicles that are on the same road. If

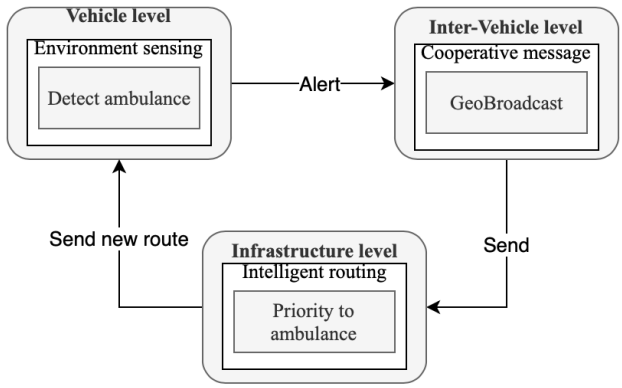



Fig. 3. A simple C-ITS use-case in the infrastructure, presented in Fig. 2.

one of these vehicles reaches a Road Side Unit (RSU), it can notify the TCC (Traffic Control Center). The C ITS-S (Central Intelligent Transportation System Station) then gives priority to the ambulance by interrupting the traffic light sequence, making the way free for it. Then the infrastructure notifies vehicles in the area so they can prepare for the arriving ambulance car. This application is indisputably important in terms of healthcare services. According to the publication of Russel G., there are on average 10 minutes of delay caused by traffic congestion in Alabama for emergency medical services. Even this small amount of time can play a critical role in the hospitalisation of patients [31] .

The mentioned example demonstrates how different layers collaborate to implement a complex smart transportation based service. The infrastructural layer is responsible for determining the new routes for the vehicles, but detecting the ambulance vehicle is out of its responsibilities. (The detection of the priority vehicle is based on the vehicle level in this example Fig. 3. In other use-cases, the ambulance may notify the infrastructure.) It is important to notice that every layer must satisfy its own responsibility in order to implement this service. Applications in each layer have their own role and responsibility and they communicate with applications on other layers. This highlights the importance of our categorization. This architectural approach is frequently used in complex systems, such as the OSI model prevalent in computer networks.

## IV. C-ITS APPLICATIONS

Section II briefly summarised how machine learning is used to solve a wide variety of problems. The following sections review novel C-ITS applications that are based on machine learning for every hierarchical layer. The listed articles that proposes a new solution are listed in Table I.

### *A. Vehicle level*

Vehicle level applications operate within the vehicle. The primary object of these is to measure the condition of ITS and the current state of its environment. ITS measure and gather information, which they then process and share the results with other participants. This layer is fundamental in

every category of C-ITS applications that are demonstrated in Section III. However applications in this layer do not directly play role in the cooperativeness of C-ITS applications, they often rely on the results of vehicle level applications. This paper classifies vehicle-level applications into 3 categories: machine-level sensing, driver-level sensing, and environment-level sensing.

*1) Machine-level:* The recent development of sensor technology has enabled to collect information about the mechanical status of the vehicle. The goal is to get a clear view of the vehicle's condition. These applications can contribute to detect vehicle fault. Based on these applications the infrastructure can be notified if a vehicle has improper or dangerous conditions. Other drivers can be alerted about a vehicle with faulty brake or one that has broken down and stopped in the middle of the road. By sharing information about the failure, other vehicles can be prepared for the unexpected event, thus preventing unwanted situations and increasing road safety.

The issue of **Maintenance prediction** has received considerable attention.

Thanks to the recent sensorisation trend in the car industry, it has become possible to measure the state of several subsystems in a car. The primary object of this is to notify the owner when maintenance is required. There are determined maintenance periods that can be scheduled based on given intervals, such as the age of the part or the mileage. Condition-based maintenance needs to be conducted based on the condition of the given part of the vehicle. While most parts, such as the timing belt and the clutch, have planned, given maintenance periods, unexpected faults can happen and result in expensive maintenance. To overcome this situation, faults can be recognized in their early stages before they cause fatal problems. This way the owner of the vehicle can be alerted to repair the vehicle before the fault happens. Using predictive maintenance notifications, the owner can prevent bigger faults and can save a considerable amounts of money on costly repairs. Maintenance prediction helps to optimise expenses, increase the reliability of the vehicle and minimise downtime. These optimizations are very important in the transportation sector. Implementing predictive maintenance reduces the number of broken cars, thus avoiding possible traffic congestion caused by unexpectedly stopped vehicles. It can also improve safety by identifying potential problems before they become critical and prevent accidents. Predictive maintenance can help reduce fuel consumption as well. If a vehicle's engine or transmission system is not operating at peak performance due to a faulty component, it may require more fuel to travel the distance. Burning more fuel releases more gases causing pollution, leads to higher travelling costs and inhaling more gas has negative health effect. Predictive maintenance has been the subject of many survey papers [32], [33], [34], [35].

**Braking system** is one of the most important parts of a vehicle, its proper functioning is crucial for the safety, and therefore, braking system fault detection has been studied extensively. A vehicle with a faulty braking system can brake more slowly or not at all. This leads to dangerous situations. Other vehicles can be real-time alerted about the arrival of a dangerous vehicle with brake problems. This work observes braking systems based on vibration monitoring, thermal imaging and oil particle analysis. This allows prepared drivers to handle the situation with appropriate caution, which increases road safety.

An extensive summary of machine learning based break fault diagnosis has been carried out by Manghai [36].

Ryan M. et al. [37] showed a digital twin simulation (DTS) for generating training data for brake fault detection systems. DTS is a virtual replica, a simulation of a physical system and allows researchers for testing and optimizing it.

Joshuva et al. [38] have introduced a classification-based system to identify brake faults. In their work data was directly collected from a Ford EcoBoost model's brake vibration measured by an accelerometer paired with an integrated charging amplifier and a signal conditioning module. They compared C4.5 and Logistic Model Tree (LMT) algorithms with 7 selected features. C4.5 is a decision tree based classification algorithm [39]. LMT builds a tree of logistic regression models to predict the class-probability for a new data point. It produced 92.75% accuracy at early fault detection [38] . In his further publication he proposed a random tree classifier that can produce 98.5% accuracy on the same dataset measured on the Ford [40].

One of the most accurate algorithm is developed by Zhang et al. [41]. Their fault detection system performed 100% accuracy and it is based on a modified convolutional neural network (CNN) and a support vector machine (SVM). CNNs are widely used in AI based applications, they are deep learning models that can learn spatial hierarchies of features from the high dimensional input data using convolutional layers. SVM is a commonly used ML algorithm to classify data in high-dimensional feature spaces. It finds the optimal hyperplane to separate the different data classes. Their study uses the disc brake simulation bench of China University of Mining and Technology as data source. This simulation contains the raw signal of friction factor and friction surface temperature of several types of breaking discs. Their work uses both statically monitored object measurements, such as the friction of the braking system parts, and the temperature of braking surfaces and uses dynamically extracted features as well.

Monitoring braking systems is not only useful for fault detection. Braking intensity can highly influence the success of a race vehicle. Federico et al. [42] studied the optimization of carbon brakes in MotoGP. The study revealed that braking efficiency is influenced by three primary factors: disc temperature, the relative speed between the disc and pads, and the amount of applied normal force. To identify the relationship between the braking efficiency and the parameters, SVM, decision trees and neural network models were used. The analysis has been proceeded on simulation generated data and concluded that the increasing temperature of a braking system can reduce intensively the braking power [42].

**Suspension** systems play an important role in the dynamics and handling of a vehicle. They are designed to reduce the vibration of the vehicle caused by the road. Fault in the suspension system can reduce the vehicle's controllability. A vehicle that is difficult to control is also a danger to other

road users. By detecting the fault, other drivers on the road can be alerted to the arrival of a dangerous vehicle. The fault of the suspension system can be detected by the use of clustering based algorithms [43]. This data-driven solution detects faults without explicitly outpointed fault features using unsupervised clustering to detect anomalous clusters based on the spring coefficient of the observed suspension systems [43]. The proposed method uses dimension reduction to determine whether a new fault occurred and the Fisher discriminant analysis to classify the data points into fault categories [43].

*2) Driver-level:* The behaviour and driving style of drivers play crucial role in transportation safety. A careless, dangerous driver can pose significant risk to others on the road. The goal of driver-level applications is to monitor the actions and behavior of the driver. Based on these, C-ITS services can alert drivers real-time when a dangerous driver occurs and prepare them to handle the dangerous driver with increased attention.

**Sleepy driving**, also known as driving while drowsy, is a highly dangerous but common condition among drivers [44]. According to Kalsi [45], sleepiness is the most significant risk factor for drivers. When drivers are tired, their reaction times can increase by more than two-fold. For instance, at a speed of 50km/h, if the aware reaction time is one second between sensing and acting, drowsiness can add an extra 14 meters before a driver can react. Moreover, Vogelpohl [46] claims that drivers in autonomous vehicles tend to lose focus faster and and take longer to regain their concentration. Alerting the driver to take a break reduces the risk coming from unfocused driving. In recent years, most research on detecting sleepiness has emphasized the use of machine learning techniques. Drowsiness detection methodologies use sensors to monitor a driver's cognitive state to detect signs of fatigue and drowsiness. It can also be integrated with other C-ITS systems, such as traffic management and emergency services, to provide real-time alerts to other road users and authorities about the presence of an unaware driver on the road. This can help prevent accidents and improve road safety for all users.

One way to detect drowsy driving is by analyzing the facial expressions of drivers. In Nietjet's work [47] a Raspberry Pi based hardware detects the tiredness of facial expressions. The system considers the eye closure ratio as the main indicator of drowsiness and detects it using a Haar cascade classifier. It is common used in face detection applications. The main essence of Haar cascade classifier is to detect and identify parts of an image that are relevant to the face (such as edges, lines).

The usage of Haar cascade classifier occurs also in the work of Byrnes and Sturton [48]. The used data has been directly provided by 23 recruited participants. The work shows that there is a change in the eye behaviour when the observed person feels drowsy (eye closure is higher, blinking becomes more frequent).

Jabbar et al. [49] developed a lightweight system that can detect drowsiness by analyzing the face of drivers and can reach up to 83% accuracy in finding drowsiness. This method also relies on the state of the driver's eye, identified by the Viola-Jones algorithm. Then a CNN extracts the features and a SoftMax layer classifies the samples into binary (drowsy/awake) categories. The Viola-Jones algorithm is a widely used object detection method. Its basic principle involves a cascade of simple classifiers, which are combined to form a more complex classifier. Each simple classifier is constructed using a Haar-like feature, which is a rectangular window that measures the difference in intensity between adjacent sub-regions of an image. The training data used by Jabbar et al. is from the National Tsing Hua University (NTHU) Driver Drowsiness Detection Dataset [50] which contains 4157 images from 4 different persons with open and closed eyes.

Transfer learning (VGG-16, VGG-19) and fully designed neural network based approach has been introduced by Hashemi et al. [50]. Transfer learning is a technique where a pre-trained model is used as a starting point for the domain specific training. This allows the model to rely on pre-existing knowledge letting it learn more quickly and effectively. VGG-16 and VGG-19 are two deep convolutional neural network models. They are widely used in computer vision tasks because they perform well on image classification and object detection tasks [51]. Their work uses the ZJU Eyeblink Database and with another 4 models added by the author and results in a 98.15% accuracy.

Guo and Markoni [52] proposed a concept that detects drowsiness real-time based on CNN and LSTM. With the use of LSTM the proposed work uses not just a picture frame at a time to detect drowsiness, but frame sequences, thus analyzing the change of the face. LSTM (Long Short-Term Memory) is a recurrent neural network that can model sequential data. It uses memory cell and captures long-term dependencies in the data. The system has been tested on the public drowsy driver dataset from ACCV 2016 competition. It contains images of people sitting in front of a simulated steering wheel and acting like they are driving. The proposed Time Skip Combination Long-Short Term Memory has outperformed most models with a 84.85% accuracy. Other methodologies are summarised here: [53]. **Driver identification** and profiling has a great amount of literature. The driver can be identified in 2 minutes of urban area driving with a 85% success probability based on their pedal handling [54]. Yang et al. [55] proposed a deep learning-based solution that can identify drivers in 10 seconds with 83% accuracy by observing several patterns of measures in the car, such as the following distance, the usage of the turn indicator, acceleration, etc. In their work they used a driving pool containing 51 drivers. Furthermore a single turn can identify a driver from 64 different drivers with more than a 50% accuracy [56]. These publications support the fact that our driving style is unique, and drivers can be identified, but the scalability of these works has not been investigated.

Once a driver is identified a user profile can be assigned to them. This profile carries information about their driving characteristics. Recently there has been an increasing interest in **user behaviour characteristic identification** [57], [58]. It can enable real-time monitoring of driver behavior and provide warnings to the driver and other road users in case of dangerous situations. It can also allow for more accurate risk assessment, which can lead to fairer and more personalized insurance pricing.

The goal of a well cited article written by Shahverdy et

al. [59] is to classify drivers into 5 driving style category (safe, drunk, drowsy, inattentive, aggressive) based on driving features, such as acceleration, gravity, throttle, speed, and used revolutions per minute. The problem is addressed in this work as an image classification task. A 2D CNN performs the classification based on visualisation, which is constructed by the help of the measured driving signals. The data was gathered by observing 3 drivers who imitate the above listed different driving styles.

Ferreira et al. [60] shows that the data for a user behaviour monitoring system can be collected from smartphone sensors. In this work the data is collected from 4 different vehicles using smartphones in the cars. The goal of this study is to identify maneuvers such as aggressive lane changing, turning, braking, accelerating and normal events. The gyroscope and the accelerometer sensors of the phone provided the data. They have compared SVM, RF, MLP, and BN models. Bayesian Network (BN) model is a probabilistic model that represents a set of variables and their dependencies using a directed acyclic graph which is used for decision making and prediction. The best performing model is Random Forest with 80% accuracy.

Lowering insurance fees for drivers who comply with traffic rules can serve as a good motivation to encourage normal driving. This approach is beneficial for both drivers and insurance companies since aggressive driving can cause accidents and create additional expenses.

AutoCoach is an intelligent agent developed by Marafie et al. [61] that implements a pervasive system to improve driver's behaviour. The system classifies drivers based on their driving behaviour and calculates a risk score based on the actions they have performed. A Support Vector Machine is used in this work to classify actions into 3 risk category. The training dataset is collected from a Toyota and BMW car. If the risk score crosses a defined threshold, the algorithm classifies the driver's behaviour as a bad one. The agent provides feedback on the current and the historical driving style.

There is extensive research on Usage Based Insurance (UBI) services as well. Yin and Chen have developed a framework that calculates the driving risk [62]. A common approach to implement UBI is to identify the driver behaviour characteristic and based on that provide an insurance pricing model [63], [64].

*3) Environment-level:* Environment-level applications measure the current state of the environment through sensors and use the obtained data to gain insights. ITS utilize this data to provide details about their environment to others. C-ITS applications share the information extracted from the raw data measured by the sensors in each system, rather than the data itself. This section aims to identify machine learning use-cases related to environment sensing.

**Vehicle detection** is an important part of environment sensing. In his study, Abbas [65] provides a comprehensive review of vehicle detection techniques. Vehicle detection is often the base of overtaking assistance, congestion prediction C-ITS applications.

Vehicle detection from Unmanned Aerial Vehicle (UAV) is studied widely due to its potential applications, such as traffic management, vehicle tracking, etc. The detection must be done precisely and fast. A detailed review from Bouguettaya et al. [66] shows how AutoEncoders, Generative Adversarial Networks, Recurrent Neural Networks and Convolutional Neural Networks are used for vehicle detection performed by UAVs.

The license plate is a unique identifier of a vehicle. Many solutions have been proposed for recognizing license plates. Fast and precise vehicle identification makes it possible to measure accurate travel time on public roads. This can contribute to several applications, such as traffic congestion detection or dangerous vehicle tracking.

In his literature review Tang [67] shows, how novel works perform at License Plate Recognition.

Motorcycle is a quick and compact way of travelling in cities with high traffic load. However, wearing proper protection is indispensable. Dasgupta et al. [68] introduced a CNN-based algorithm that first identifies a motorcycle, then crops the head of the driver (and passenger) and detects **helmet usage** with 96.3% accuracy. ITS can detect motorcyclists driving irresponsibly without helmets and alert the relevant authorities. This will increase the regulation of the wearing of protective equipment, resulting in safer driving.

**Forward Collision Warning** (FCW) is an important and valuable application that has received a lot of attention in research. According to statistics, the use of FCW reduces front-to-rear crash rates by 27% and front-to-rear injury crash rates by 20% [69]. FCW was introduced first in 2000 by Mercedes-Benz, but today, almost every car manufacturer offers it as an optional extra.

**Autonomous Emergency Braking Systems** (AEB) are triggered by FCW. When the vehicle detects the possibility of a collision, it executes an emergency braking maneuver to prevent the crash. In the U.S. the front-to-rear-crash rate has been reduced by 50% and front-to-rear injury crash rate by 56% thanks to the usage of FCW and AEB % [69]. Tawfeek and El-Basyouny [70] developed a novel FCW method and compared it with 6 existing FCW algorithms. This comparison showed that the proposed work can overperform every existing algorithm on every speed level. In their work, they defined a custom warning distance function and selected the important parameters to use in the function using linear regression. These warnings can be propagated on the road and can prepare the following vehicles for sudden braking.

Different lanes allow different maneuvers. A dashed lines allows vehicles to perform overtaking and lane changing, while a continuous white line prohibits such actions. FCW must consider the lane type and recognize incoming and oncoming traffic. Overtaking assisting C-ITS applications also need to be able to recognise the traffic lanes in order to support or reject overtaking. Song et al. [71] approach this problem using stereo vision processing. Stereo vision processing extracts information from 3D visual data. Using 3D data provides added value to the system, as it makes it possible to extract relative distance between the cameras and the object, as well as to correct the perspective distortion. They present a CNN based FCW system that classifies current and adjacent lanes with 100% accuracy [71].

The study by Tang et al. [72] is one of the first publications to review lane detection systems.

They claim that camera or lidar sensor-based lane detection can achieve up to 96% accuracy. This result was achieved by Zou et al. [73]. They combined a CNN with an LSTM to detect lanes in a continuous driving scenario. They evaluated the developed model on TuSimple dataset that contains almost 4000 image sequences of driving.

Traffic signs are the most common form of regulation in traffic. They indicate what actions are prohibited, allowed, or limited to drivers. However, distracted drivers may overlook traffic signs, leading to dangerous situations. If a vehicle can recognize traffic signs, it can continuously display them to the driver or, in autonomous cases, the vehicle can follow the rules indicated by traffic signs. Vehicles can also share recognized traffic signs to inform, help or alert others.

The first review on **Traffic Sign Recognition** (TSR) was published by De la Escalera et al. in 2003 [74]. It is noticeable that back in that time they claimed the number of possible solutions was highly limited. They investigated a NN based solution that achieved 80% accuracy in average on several signs. The significant improvement in neural network technologies radically increased the performance of TSR. Novel methodologies can achieve performances up to 95% and can reach even 100%. This shows how the spread of machine learning contributes to the efficiency of ITS applications.

Stallkamp et al. [75] surveyed a human focus group to classify traffic signs and concluded that the average human performance was 98.8% accurate. This is close to perfect, but NN-based solutions can clearly outperform human capabilities.

Rajendran et al. [76] proposed a Region Based CNN (R-CNN) method that can reach up to 99.7% accuracy on the German Traffic Sign Detection Benchmark (GTSRB) dataset. It contains 43 classes of traffic signs on more than 50.000 images [77]. R-CNNs are a subclass of CNNs. They localize and classify images by dividing them into regions and extract features from them separately.

Li and Wang [78] proposed another R-CNN solution using YOLOv3 that reaches comparable (99.6%) performance on the same dataset. YOLOv3 is a pre-trained CNN architecture, that is often used for transfer learning. This proves that CNN is a very effective and successful architecture for TSR.

### *B. Inter-Vehicle level*

This Section presents how machine learning can contribute to applications that are used inter-vehicle. These applications rely on V2V communications, and the base of them is the capability of information exchanging. The cooperativeness of applications from Section IV-A comes from sharing the measured information. Solutions provided by this section are based on the communication, the sent information and the formed networks among vehicles. The goal of these applications is to implement inter-vehicle communication and to facilitate warnings and recommendations to drivers based on collective information about a network of vehicles.

VANET is an important subcategory of Mobile Ad Hoc Networks (MANET) and is commonly used in the network layer of V2V. It provides ad hoc network solution to vehicles for ubiquitous connectivity on the road with OBUs and RSUs. VANET is a practical solution to implement V2V communication because it can differ and really flexible in numerous aspects: size, number of vehicles, speed, etc. The characteristic of VANET is highly dynamic in terms of their topology, because it has to implement high mobility, dynamic topology, frequent connection and disconnections [79]. It is an useful solution, however it suffers from many security threats: information integrity, modulation problems, traffic overload, etc. Machine learning based solutions are to provide solution to these. The following sections present machine learning use-cases in V2V communication and in VANET environments. The content of this section is summarized in Table II.

A comprehensive and detailed summary of machine learning use-cases in vehicular networks can be found in the survey conducted by Tang et al. [80].

*1) Network security level:* Ensuring the security of V2V and VANETs is fundamental. Although vehicle networks are protected by several authentication and encryption measures, fraudulent activities can still occur due to vulnerabilities in the system. To detect and prevent such attacks, machine learning methodologies are implemented to detect abnormalities and predict hacker attacks. Ensuring the continuous security of VANETs is crucial because applications rely on the data transmitted through them. Using corrupted data leads to misbehaving, causing an inefficient, moreover, dangerous transportation. VANET attacks can occur in different aspects of the network [81]:

- Availability: The network is functioning at any time and always available for nodes.
- Authenticity and identification: Network stations must be identified before using it.
- Confidentiality: The transmitted data is not readable for everyone because it can contain confidential information.
- Integrity and data trust: The transmitted data has not been changed after sending it.
- Accountability: Change of transmitted data must be transparent and accountable.

To develop useful, efficient and safe V2V applications, it is fundamental to reduce the risk coming from vulnerability. This section presents novel solutions that can increase the security of V2V. **Availability** corruption of VANETs makes the network unavailable. This is often accomplished through Distributed Denial of Service (DDoS) attack against the network. The goal of DoS is to overload the network's resources, causing delays or complete interruptions in the network's behavior. In their study, Alrehan and Alhaidari [82] provide a comprehensive overview of how machine learning can prevent DDoS attacks. Most of the surveyed works in their article focus on detecting outlying behavior in VANETs using classification algorithms such as SVM.

Kadam and Krovi [83] have developed a system based on a hybrid combination of k-NN and SVM. They evaluated their system on the Kaggle DDoS dataset and achieved up to 93% accuracy. This contains attributes of packets, such as protocol, source and destination IP and port, etc.

Polat et al. [84] use a more complex, deep learning based model to solve this problem on software-defined network based VANETs. They introduced a sparse AutoEncoder and

TABLE I
SUMMARY OF VEHICLE-LEVEL APPLICATIONS

| Author | Article | Year | Application | Used ML Algorithm |
|---|---|---|---|---|
| Manghai et al. | [36] | 2017 | Brake fault detection | SVM |
| Joshuva et al. | [38] | 2020 | Brake fault detection | C4.5, LMT |
| Joshuva et al. | [40] | 2021 | Brake fault detection | LMT |
| Zhang et al. | [41] | 2021 | Brake fault detection | SVM, CNN |
| Federico et al. | [42] | 2021 | MotoGP braking torque estimation | SVM, Decision Tree, NN |
| Wang | [43] | 2014 | Suspension fault detection | Discriminant analysis |
| Nietjet | [47] | 2018 | Driver tiredness detection | Haar cascade classifier |
| Byrnes and Sturton | [48] | 2018 | Driver tiredness detection | Haar cascade classifier |
| Jabbar et al. | [49] | 2020 | Driver tiredness detection | Viola-Jones + CNN |
| Hashemi et al. | [50] | 2020 | Driver tiredness detection | VGG-16, VGG-19 |
| Guo and Markoni | [52] | 2019 | Driver tiredness detection | CNN+LSTM |
| Remeli et al. | [54] | 2019 | Driver identification from driving style | CNN+LSTM |
| Yang et al. | [55] | 2021 | Driver identification from driving style | CNN |
| Shahverdy | [59] | 2020 | Driving style classification | 2d CNN |
| Ferreira et al. | [60] | 2017 | Driving style identification from smartphone sensor | SVM, RF, MLP, BN |
| Marafie et al. | [61] | 2019 | Driving behaviour management | SVM |
| Yin and Chen | [62] | 2018 | Usage-based insurance | AdaBoost |
| Dasgupta et al. | [68] | 2019 | Motorcycle helmet detection | CNN |
| Tawfeek and El-Basyouny | [70] | 2018 | Forward collision warning | Regression analysis |
| Song et al. | [71] | 2018 | Lane detection and classification | CNN |
| Zou et al. | [73] | 2019 | Lane detection | CNN+LSTM |
| Rajendran et al. | [76] | 2019 | Traffic sign recognition | YOLOv3 |
| Li and Wang | [78] | 2018 | Traffic sign recognition | YOLOv3 |

SoftMax classifier-based neural network and evaluated it on simulated traffic flow data. Their model achieved 96.9% accuracy in detecting DDoS attacks.

When a corrupted node in the network refuses to forward packages, thus blocking the communication is called a blackhole attack. This can disrupt the operation of the network by making it unavailable for applications.

Acharya and Oluoch [85] demonstrate a method in which the combination of statistical approaches and SVM can help identify corrupted nodes and detect blackhole attacks. Their system finds blackhole attacks with over 98% accuracy in VANETs. Similarly, Pandey and Singh [86] use SVM to detect nodes performing blackhole attack with 95% accuracy by observing their energy consumption. An unsupervised, clustering based anomaly detection is performed in the work of Farahani [87]. In this work k-NN model is implemented to identify fraud clusters in the latent space of nodes with high confidence.

**Providing authenticity and identity** is an important task in VANETs. To operate correctly, data correctness must be guaranteed. Attacks against authenticity can harm transmitted data correctness, causing congestion or danger in traffic. A common and dangerous form of attack is location spoofing.

In their study, So et al. [88] use k-NN and SVM to classify nodes that perform attacks against the network. They evaluate their approach on the Vehicular Reference Misbehavior (VeReMi) dataset, a labeled simulated dataset containing several attack types [89]. This approach achieves 81.6% accuracy in finding malicious nodes.

Sharma et al. [90] propose a machine learning based framework that can classify several types of position falsification: fixed position transmitting, added offset to actual position or transmitting random positions. The framework was also evaluated on the VeReMi dataset, and performed with 98.9% accuracy

Sybil attacks allow hackers to show non-existing nodes to overrule decisions by faking the state of the architecture. The proposed work of Hammi et al.[91] uses a resource testing approach that is built on the fact that each node has limited computational resource. If a node presents multiple entities, their computation testing will indicate that. For the classification they used LR, SVM, RF, NN. The proposed classification method can recognize several sybil scenarios and reaches 86% accuracy on a simulated dataset.

Kamel et al. [92] use a different approach based on OBU's and RSU's misbehaviour reports. The proposed system is distributed into 3 phases: general misbehaviour type detection to detect the type of misbehavior, pseudonym linking to link pseudonyms coming from the same vehicle, sybil type detection to detect the type of sybil attack. These phases rely on AutoEncoders, RNNs, and MLP. The experimental data is simulated on the Luxembourg SUMO Traffic scenario validated by the University of Luxembourg. The proposed complex system performs 95.6% accuracy.

**Confidentiality** harming attacks compromise the privacy of nodes by making private messages accessible to hackers. Eavesdropping is a passive attack wherein hackers intercept information by listening in on VANET communications without permission. Detecting eavesdropping attacks can be challenging due to their passive nature.

Rahal et al. [93] introduced a process for botnet detection in VANETs that monitors vehicle and in-vehicle activity to detect DDoS attacks and Eavesdropping. The proposed work uses ML based algorithms (SVM, k-NN) and performs 99.4% accuracy at finding botnets.

Hackers compromising **integrity and data trust** modify packets in transit, thus altering their content before they are received by the intended recipient. A common attack form is broadcasting again messages that have already been broadcasted to manipulate the state of the network to gain benefits. These attacks are called replay attacks.

Fan et al. [94] showed that SVM can accurately detect

TABLE II
SUMMARY OF INTER-VEHICLE LEVEL APPLICATIONS

| Author | Article | Year | Application | Used ML Algorithm |
|---|---|---|---|---|
| Kadam and Krovi | [83] | 2021 | Detect DDoS in VANET | Hybrid k-NN+SVM |
| Polat et al. | [84] | 2020 | Detect DDoS on SDN-based VANET | AutoEncoder |
| Acharya and Oluoch | [85] | 2021 | Detect blackhole attack | SVM |
| Pandey and Singh | [86] | 2020 | Detect blackhole attack | SVM |
| Farahani | [87] | 2021 | Detect blackhole attack | k-NN |
| So et al. | [88] | 2018 | Node misbehaviour detection in VANET | k-NN, SVM |
| Sharma et al. | [90] | 2021 | Location spoofing detection | k-NN, Decision Tree, RF |
| Hammi et al. | [91] | 2022 | Sybil attack detection | LR, SVM, RF, NN |
| Kamel et al. | [92] | 2019 | Sybil attack detection | AutoEncoder, RNN, MLP |
| Rahal et al. | [93] | 2022 | Botnet detection | SVM, k-NN |
| Fan et al. | [94] | 2016 | Routing replay attack detection | SVM |
| Ye et al. | [95] | 2019 | Resource allocation | Q-learning |
| Gao et al. | [96] | 2019 | Resource allocation | NN |
| Morocho et al. | [97] | 2020 | Extend wireless reachability in V2X | Encoder-Decoder NN |
| Moreira et al. | [98] | 2020 | QoS predictability in V2X communcation | RF |
| Mo et al. | [99] | 2018 | Overtaking assistant | Decision tree |

replay attacks on a simulated communication scenario with 99.95% accuracy by analyzing the MAC Layer information of packets sent in network.

**Accountability** is mostly provided by implementing encryption technologies, using machine learning is not a common approach to provide the accountability of VANETs.

*2) Network efficiency level:* As discussed in Section IV-B1, VANETs have a dynamic nature, which makes effective resource allocation critical for ensuring availability. Machine learning can play a vital role in supporting resource allocation tasks.

The paper of Ye et al. [95] is one of the most cited publications on the topic. They present a novel deep reinforcement learning (Q-learning) based method for decentralized resource allocation. They claim that their algorithm allows each node and link to learn how to make resource allocation decision locally without global information effectively. The proposed algorithm results approximately 5% higher probability of satisfied V2V links than the defined baseline and increases the Sum Rate of V2I links by 3.5 times compared to random resource allocation method.

Gao et al. [96] approach this problem differently. They proposed a supervised learning deep neural network (DNN) based resource allocation method. To forward packages, hopping strategy is based on the shortest hop distance between nodes.

Morocho et al. [97] introduce a new, deep encoder-decoder and reinforcement learning based strategy for extending the reachability of multi-hop communications by up to 66.7%. The experiments in these works were tested on end-to-end IEEE 802.11p simulation.

Applications using V2X technology have stringent Quality of Service (QoS) requirements. Each application requires different network QoS. If netwok parameters do not satisfy the requirements of the application, its proper working is not guaranteed. The proposed work of Moreira et al. [98] proposes a machine learning based method to predict the QoS of a V2V network. In the proposed work, several supervised algorithms are evaluated, but the best performing is Random Forest that reaches up to 85% accuracy.

*3) Application level:* Applications that operate between a group of vehicle to perform synchronised actions are the base of cooperative driving. The aim is to enable vehicles to make decisions collectively, rather than independently, often by using summarized individual measurements. In this section, we cover the most common cooperative driving applications.

Cooperative Adaptive Cruise Control (C-ACC) is an application that allows vehicles to follow each other cooperatively. Using C-ACC gives the ability for vehicles to perform platooning, that is commonly referred as “driving together”. By continuously communicating with each other, vehicles can reduce the distance between them, which would not be possible without cooperation due to human reaction delay. The minimum following interval is typically 0.5 sec, compared to non-cooperative ACC where it is 1.6 sec [100]. Using C-ACC causes greater fuel economy, reduces congestion and lower the occupancy of roads [100]. This application mainly uses CAM and DENM messages to communicate hazardous road situations and to share vehicle parameters between convoys [100].

Cooperative overtaking assistant (COA) helps drivers to decide whether to perform overtaking maneuver on roads with bad visibility.

In their work, Mo et al. [99] introduce an overtaking assistant system and based on simulated experiments it increases the safety of overtaking. The overtaking process is built on multiple decision trees.

Traditional overtaking systems rely on processed visual data, but their work takes advantage of V2X. Strunz et al. [101] demonstrate how COA can contribute to platooning on freeways, increasing the average speed by 36%.

Other applications that are based on cooperativity [102]: intersection movement assist, left turn assist, cooperative forward collision avoidance, and emergency electronic brake light.

### C. Infrastructure level

Infrastructure level applications operate on the “big picture”. Their goal is to make the whole infrastructure effective, fast and safe. Smart city is the concept where sustainability and efficiency is provided by the support of technology. Smart cities

implement solutions for a great variety of their utilities, such as health care, infrastructure, water supply, pollution reducing, and transportation. Smart transportation management systems in smart cities integrate novel C-ITS technologies to improve constructional, management and support operations for ITS. The papers presenting solutions are listed in Table III.

The limited space and continuously increasing number of vehicles in big cities make parking a real challenge. Searching for parking lots, especially during peak hours, can waste a great amount of time and fuel, hold up traffic, and increase emissions. **Parking space** inventory control systems have been developed to provide solution to this difficulty.

Shin et al. [103] propose a neural network based method to enhance the effectiveness of intelligent parking guidance systems in parking areas where the necessary IoT devices are installed. This robust and lightweight method is capable of managing both public and private parking spaces in cities. The proposed work uses the current position of the vehicle and a centralized database about parking lots.

The work of Liu et al. [104] predicts the available parking lots with a neural network based approach. This work reached high accuracy based on the experimental results.

Priya et al. [105] propose an intelligent parking system that calculates the best-fit and optimal parking area based on size. The system extracts the dimensions of the vehicles using image processing CNN. Their work can achieve up to 85% accuracy in extracting proper dimensions.

Yang and Lam [106] survey how the implementation of intelligent parking services can create benefits for drivers.

Traffic congestion often forms in crowded areas. Spending more unexpected time on the road has a bad effect on travelling or transportation. Machine learning can help **predict the delay** of transportation.

Traffic congestion recognition and forecasting can be based on several data sources. Using computer vision and image processing for congestion detection is extensively studied. In their work, Jian et al. [107] show that roadway congestion detection can be done effectively using recordings of UAVs, as the "Electronic Eyes in the Skies". They use convolutional neural network (CNN) to extract information from the captured images. They evaluated the system on a dataset that contained 8000 images and reached 93.5% accuracy. Bisio et al. [108] have published an extensive review of drone-based traffic monitoring systems. The work of Liu et al. [109] also uses computer vision to detect congestion, but their work relies on large and medium cities' public security cameras and their algorithm reaches up to 82.5% accuracy. Gatto and Forster [110] use audio data from roads to determine the congestion. They extract 13 MFCC coefficients from the sound files and classify them using Random Forest Classifier. Their work reaches 95% accuracy on average on data collected from YouTube traffic videos.

Leung et al. [111] propose a heterogeneous approach to predict trolley bus delay based on transit and weather data. The presented method takes into account the current weather condition and uses fuzzy-logic based machine learning. Rather than using binary labels as the output for a decision it is capable of reasoning and uses a value between 0 and 1. The case study in this work, evaluated on Toronto Transit Commission database shows that this approach can approximate the trolley bus delay with 3.7 minutes mean prediction error.

Wu et al. [112] presents a method to predict the delay of urban railways. They present a novel Convolution LSTM Encoder-Decoder neural network based approach to predict delay. Their model using pure LSTM network reaches 16.82 sec mean prediction error, that is remarkably accurate.

An extensive amount of studies can be found on airplane arrival time delay prediction [113], [114].

Traffic prediction is also important in the aspect of infrastructure. **Predicting traffic** helps to react proactively to possible events in the future, thus preventing them. Proper and accurate prediction can forecast possible congestion, thus letting the smart routing applications avoid them. Nagy and Simon [115] give a comprehensive and detailed survey about novel traffic prediction methodologies.

Using graph convolution networks (GCN) has gained a big momentum at predicting traffic. Graph neural networks are capable of capturing rich feature of non-euclidean structured data and preserve its spatial structure. The paper written by Zhao et al.[116] is a frequently cited article that shows how real-time traffic forecasting can be performed using temporal GCN (t-GCN) and GCN. Other methodologies also use GCN to predict traffic [117].

RF and ARIMA, -which is a statistical approach-, are commonly used algorithms for traffic prediction, however, complex models, such as GCN tend to overperform it [117].

Other solutions published in the last five years mostly use LSTM and CNN [118].

High traffic affects the network layer as well by overloading it. Zhao et al. [119] proposed a deep reinforcement learning-based system that **optimizes network usage** and load for crowd management in smart cities. The proposed system uses deep Q-network model. The experiments are evaluated on a simulated environment from Topology Zoo, NSFCNET in Beijing, China. Their work has succeeded at optimizing network load and service availability.

One area where such possibilities can be useful is emergency management, which is expected to play a crucial role in the evolution of modern cities. Emergency vehicles, including ambulances, fire trucks, police cars, and transit agents' vehicles, need to be quickly assigned to respond to critical situations as soon as possible.

**Vehicle Routing Problem** (VRP) is one of the most intensively studied optimization problem for which sufficient models and algorithms have been proposed. Emergency management reduces negative unfortunate or critical events. Travel time of EVs is a critical aspect of health care. Smart routing systems (SRS) can help to reduce this time by optimizing emergency vehicle routing and reducing hospitalization time. The survey of Tassone and Choudhury [120] shows that solving VRP helps to reduce the travelling and response time, and minimize the total cost of hospitalization. It details classical approaches and used algorithms and compares their performance.

Bai et al. [121] let the reader understand how machine learning can contribute to VRP and survey novel ML based

TABLE III
SUMMARY OF INFRASTRUCTURE-LEVEL APPLICATIONS

| Author | Article | Year | Application | Used ML Algorithm |
|---|---|---|---|---|
| Shin et al. | [103] | 2018 | Intelligent parking guidance system | DNN |
| Liu et al. | [104] | 2020 | Parking lot availability | DNN |
| Priya et al. | [105] | 2019 | Optimal parking | CNN |
| Jial et al. | [107] | 2019 | Congestion detection using UAVs | CNN |
| Liu et al. | [109] | 2020 | Congestion detection using public cameras | CNN |
| Gatto and Forster | [110] | 2021 | Congestion detection using audio features | RF |
| Leung et al. | [111] | 2020 | Trolley bus delay prediction | Fuzzy-logic |
| Wu et al. | [112] | 2019 | Urban railway delay prediction | C-LSTM+Encoder-Decoder |
| Zhao et al. | [116] | 2019 | Real-time traffic forecasting | t-GCN |
| Zhao et al. | [119] | 2019 | Routing management | Q-Learning |
| Hussein et al. | [122] | 2022 | Ambulance vehicle routing | DNN |
| Hussein et al. | [123] | 2022 | Ambulance vehicle routing | BA-CNN |
| Nallaperume et al. | [125] | 2019 | Smart traffic control | Q-learning |

methodologies.

Hussein [122] investigates how machine learning can help find the shortest travelling time for ambulance vehicles (AV). In their first ML based approach, they developed a neural network based model that can suggest optimal travelling path based on information about the accident, the injured patients and on position describers. In their simulation, the proposed approach could reduce the travelling distance by 3 times with negligible computing time [122]. In their another publication [123], a Bat Algorithm-Based Convolutional Neural Network (BA-CNN) is proposed. Bat is one of the most powerful optimisation algorithm that is inspired by the echolocation behavior of bats proposed by Yang and Gandomi [124]. The BA-CNN is very effective and can outperform existing methodologies [123].

Nallaperume et al. [125] introduce a novel framework that can implement smart traffic control based on multiple data source, such as social media data, IoT, weather data. The presented framework analyses the emotional status of drivers using social media data. This methodology uses deep reinforcement learning to improve traffic flow, reducing average waiting time at traffic lights by making traffic signal control decisions based on processing real-time data streams.

Smart traffic light control is a fundamental part of infrastructure optimization. Reinforcement learning is a common approach, but different works have different reward definitions. Rewarding scores can be based on reducing queue length [126], waiting time [127], or improving throughput and signal frequency in traffic lights [128].

## V. CHALLENGES

Reducing computational time is a major challenge in developing ML-based C-ITS applications that can outperform existing solutions based on classical approaches. Evaluating complex NN architectures, slow data processing, or using ML for problems with high-dimensional state space representation can be time-consuming. In rapidly changing traffic, the response time of these applications must be minimized.

Ensuring that these applications are reliable enough to be implemented in real-world scenarios is also of paramount importance. Even if the application can achieve impressive accuracy, rare misbehavior can cause fatal issues.

If safety is provided, it is truly important to gain confidence and trust from customers. The real implementation of these application is depending on the acceptance and trust, thus the demand of users.

The accuracy of these applications depends largely on the quality and quantity of the input data. Researchers need to use accurate, reliable and correct data, which are difficult to collect and very expensive to buy from various data providers.

However, between all the listed problems, the most pressing is scalability. The applications surveyed in our article operate most of the time in simulated environment, or on a pilot area. For example, forecasting traffic congestion propagation is much more difficult on a complex city architecture.

## VI. CONCLUSION

This review presented novel C-ITS use-cases that utilize machine learning. Firstly, a comprehensive introduction to machine learning and C-ITS technologies is provided to help readers understand their potential. Next, the paper presents novel C-ITS applications using machine learning. The categorization of the applications presented in this paper were developed by us, and we believe they enhance the clarity and structure of the paper. The paper shows that C-ITS applications are built on vehicle-level, inter vehicle-level and infrastructure-level applications as well.

The study explains that each hierarchical level has its own responsibility in accomplishing complex solutions, and the cooperation of applications on different levels is indispensable for the implementation of smart transportation. Machine learning has a huge impact on almost every aspect of C-ITS applications and can radically improve the efficiency, speed, and performance of these applications. The article highlights the need for further research and development in this field to overcome challenges such as reducing computational time, gaining customer confidence and solving scalability issues. Using ML based technologies has a huge potential to develop novel C-ITS applications.

We hope that our article will help the reader understand the importance of machine learning in C-ITS applications and give a summary of how this area builds up.

## ACKNOWLEDGMENT

The research reported in this paper is part of project no. BME-NVA-02, implemented with the support provided by the Ministry of Innovation and Technology of Hungary from

the National Research, Development and Innovation Fund, financed under the TKP2021 funding scheme.

## REFERENCES

[1] "Vehicles in use europe 2022 report." ACEA, 01 2022.

[2] D. Lee, D. Fahey, A. Skowron, M. Allen, U. Burkhardt, Q. Chen, S. Doherty, S. Freeman, P. Forster, J. Fuglestvedt, A. Gettelman, R. D. León, L. Lim, M. Lund, R. Millar, B. Owen, J. Penner, G. Pitari, M. Prather, R. Sausen, and L. Wilcox, "The contribution of global aviation to anthropogenic climate forcing for 2000 to 2018," *Atmospheric Environment*, vol. 244, p. 117834, jan, DOI: 10.1016/j.atmosenv.2020.117834.

[3] *The Future of Rail*. OECD, feb 2019, DOI: 10.1787/9789264312821-en.

[4] P. Lasley, "2021 urban mobility report," 2021.

[5] M. Zuurbier, G. Hoek, M. Oldenwening, V. Lenters, K. Meliefste, P. van den Hazel, and B. Brunekreef, "Commuters' exposure to particulate matter air pollution is affected by mode of transport, fuel type, and route," *Environmental Health Perspectives*, vol. 118, no. 6, pp. 783–789, 2010, DOI: 10.1289/ehp.0901622.

[6] S. Dick, "Artificial Intelligence," *Harvard Data Science Review*, vol. 1, no. 1, 07 2019, DOI: 10.1162/99608f92.92fe150c.

[7] S. I. for Human-Centered AI, "The ai index 2022 annual report," march 2022.

[8] A. Althnian, D. AlSaeed, H. Al-Baity, A. Samha, A. B. Dris, N. Alzakari, A. Abou Elwafa, and H. Kurdi, "Impact of dataset size on classification performance: an empirical evaluation in the medical domain," *Applied Sciences*, vol. 11, no. 2, p. 796, 2021, DOI: 10.3390/app11020796.

[9] J. Prusa, T. M. Khoshgoftaar, and N. Seliya, "The effect of dataset size on training tweet sentiment classifiers," in *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2015, pp. 96–102, DOI: 10.1109/icmla.2015.22.

[10] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.

[11] M. Feurer and F. Hutter, "Hyperparameter optimization," in *Automated machine learning*. Springer, Cham, 2019, pp. 3–33.

[12] M. Hossin and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International journal of data mining & knowledge management process*, vol. 5, no. 2, p. 1, 2015, DOI: 10.5121/ijdkp.2015.5201.

[13] S. B. Kotsiantis, I. Zaharakis, P. Pintelas et al., "Supervised machine learning: A review of classification techniques," *Emerging artificial intelligence applications in computer engineering*, vol. 160, no. 1, pp. 3–24, 2007.

[14] R. Xu and D. Wunsch, "Survey of clustering algorithms," *IEEE Transactions on neural networks*, vol. 16, no. 3, pp. 645–678, 2005.

[15] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM computing surveys (CSUR)*, vol. 41, no. 3, pp. 1–58, 2009.

[16] M. Kiran, C. Wang, G. Papadimitriou, A. Mandal, and E. Deelman, "Detecting anomalous packets in network transfers: investigations using pca, autoencoder and isolation forest in tcp," *Machine Learning*, vol. 109, no. 5, pp. 1127–1143, 2020, DOI: 10.1007/s10994-020-05870-y.

[17] M. Sánchez-Rico and J. Alvardo, "Dimensionality reduction techniques as a preliminary step to cluster analysis: A comparison between pca, t-sne and umap," in *9th European Congress of Methodology*, 2020.

[18] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *Towards Data Sci*, vol. 6, no. 12, pp. 310–316, 2017.

[19] L. Chen, A. Habibovic, M. Gråsjö, M. Adebahr, and P. King, "Cloud-based traffic control: a system of systems for accelerating c-its deployment and autonomous vehicle integration," in *Virtual ITS European Congress, 9-10 November 2020*, 2020.

[20] E. Vieira, J. Almeida, J. Ferreira, T. Dias, A. V. Silva, and L. Moura, "A roadside and cloud-based vehicular communications framework for the provision of c-ITS services," *Information*, vol. 14, no. 3, p. 153, mar 2023, DOI: 10.3390/info14030153.

[21] G. Naik, B. Choudhury, and J.-M. Park, "IEEE 802.11bd &amp 5g NR v2x: Evolution of radio access technologies for v2x communications," *IEEE Access*, vol. 7, pp. 70 169–70 184, 2019, DOI: 10.1109/access.2019.2919489.

[22] S. Chen, J. Hu, Y. Shi, Y. Peng, J. Fang, R. Zhao, and L. Zhao, "Vehicle-to-everything (v2x) services supported by LTE-based systems and 5g," *IEEE Communications Standards Magazine*, vol. 1, no. 2, pp. 70–76, 2017, DOI: 10.1109/mcomstd.2017.1700015.

[23] "En 302 636-4-1 (v1.4.1) – geographical addressing and forwarding for point-to-point and point-to-multipoint communications." ETSI, 11 2019.

[24] "En 302 637-2 (v1.4.1) – media-dependent functionalities for its-g5." ETSI, 04 2019.

[25] "En 302 637-3 (v1.3.1) – specifications of decentralized environmental notification basic service." ETSI, 04 2019.

[26] "Tr 103 562 (v2.1.1) – analysis of the collective perception service (cps); release 2." ETSI, 12 2019.

[27] "Ts 103 301 (v1.2.1) – facilities layer protocols and communication requirements for infrastructure services." ETSI, 08 2018.

[28] "Ts 102 894-2 (v1.3.1) – applications and facilities layer common data dictionary." ETSI, 08 2018.

[29] "Ts 103 300-3 (v2.1.1) – specification of vru awareness basic service; release 2." ETSI, 11 2020.

[30] L. Bokor, "V2x jarmukommunikacios protokollok forgalommenedzsment fokuszu bigdata elemzeshez es ebeavatkozashoz felhasznalhato adatelemei," 2020, accessed: 2022-05-19.

[31] R. Griffin and G. McGwin Jr, "Emergency medical service providers' experiences with traffic congestion," *The Journal of emergency medicine*, vol. 44, no. 2, pp. 398–405, 2013, DOI: 10.1016/j.jemermed.2012.01.066.

[32] C. Krupitzer, T. Wagenhals, M. Züfle, V. Lesch, D. Schäfer, A. Mozaffarin, J. Edinger, C. Becker, and S. Kounev, "A survey on predictive maintenance for industry 4.0," *arXiv preprint arXiv:2002.08224*, 2020.

[33] R. Prytz, "Machine learning methods for vehicle predictive maintenance using off-board and on-board data," Ph.D. dissertation, Halmstad University Press, 2014.

[34] K. Miller and A. Dubrawski, "System-level predictive maintenance: review of research literature and gap analysis," *arXiv preprint arXiv:2005.05239*, 2020.

[35] Y. Wen, M. F. Rahman, H. Xu, and T.-L. B. Tseng, "Recent advances and trends of predictive maintenance from data-driven machine prognostics perspective," *Measurement*, vol. 187, p. 110276, 2022, DOI: 10.1016/j.measurement.2021.110276.

[36] A. Manghai, R. Jegadeeshwaran, and V. Sugumaran, "Brake fault diagnosis through machine learning approaches–a review," *Structural Durability & Health Monitoring*, vol. 11, no. 1, p. 43, 2017.

[37] R. Magargle, L. Johnson, P. Mandloi, P. Davoudabadi, O. Kesarkar, S. Krishnaswamy, J. Batteh, and A. Pitchaikani, "A simulation-based digital twin for model-driven health monitoring and predictive maintenance of an automotive braking system," in *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15- 17, 2017*, no. 132. Linko¨ping University Electronic Press, 2017, pp. 35–46, DOI: 10.3384/ecp1713235.

[38] A. Joshuva, S. Anaimuthu, N. Selvaraju, S. J. Muthiya, and M. Subramaniam, "A machine learning approach for vibration signal based fault classification on hydraulic braking system through c4. 5 decision tree classifier and logistic model tree classifier," *SAE Technical Paper, Tech. Rep.*, 2020.

[39] S. Ruggieri, "Efficient c4. 5 [classification algorithm]," *IEEE transac- tions on knowledge and data engineering*, vol. 14, no. 2, pp. 438–444, 2002.

[40] J. Arockia Dhanraj, S. Muthiya, M. Subramaniam, S. Salyan, P. Chaurasiya, A. Gopalan, and S. Anaimuthu, "A comparative study with j48 and random tree classifier for predicting the state of hydraulic braking system through vibration signals," *SAE Tech. Pap*, pp. 1–6, 2021, DOI: 10.4271/2021-28-0254.

[41] X. Zhang, M. Zhang, Z. Xiang, and J. Mo, "Research on diagnosis algorithm of mechanical equipment brake friction fault based on mcnn-svm," *Measurement*, vol. 186, p. 110065, 2021, DOI: 10.1016/j.measurement.2021.110065.

[42] F. Bonini, G. Manduchi, N. Mancinelli, and A. Martini, "Estimation of the braking torque for motogp class motorcycles with carbon braking systems through machine learning algorithms," in *2021 IEEE International Workshop on Metrology for Automotive (MetroAutomotive)*. IEEE, 2021, pp. 1–6, DOI: 10.1109/metroautomotive50197.2021.9502878.

[43] G. Wang and S. Yin, "Data-driven fault diagnosis for an automobile suspension system by using a clustering based method," *Journal of the Franklin Institute*, vol. 351, no. 6, pp. 3231–3244, 2014, DOI: 10.1016/j.jfranklin.2014.03.004.

[44] A. Moradi, S. S. H. Nazari, and K. Rahmani, "Sleepiness and the risk of road traffic accidents: A systematic review and meta-analysis of previous studies," *Transportation research part F: traffic psychology and behaviour*, vol. 65, pp. 620–629, 2019, DOI: 10.1016/j.trf.2018.09.013.

[45] J. Kalsi, T. Tervo, A. Bachour, and M. Partinen, "Sleep versus non-sleep-related fatal road accidents," *Sleep medicine*, vol. 51, pp. 148–152, 2018, DOI: 10.1016/j.sleep.2018.04.017.

[46] T. Vogelpohl, M. Kühn, T. Hummel, and M. Vollrath, "Asleep at the automated wheel—sleepiness and fatigue during highly automated driving," *Accident Analysis & Prevention*, vol. 126, pp. 70–84, 2019, DOI: 10.1016/j.aap.2018.03.013.

[47] N. Nietjet, "A proposed accident preventive model for smart vehicles," vol. 6, pp. 001–010, 02 2018.

[48] A. Byrnes and C. Sturton, "On using drivers' eyes to predict accident-causing drowsiness levels," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 2092–2097, DOI: 10.1109/itsc.2018.8569293.

[49] R. Jabbar, M. Shinoy, M. Kharbeche, K. Al-Khalifa, M. Krichen, and K. Barkaoui, "Driver drowsiness detection model using convolutional neural networks techniques for android application," in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*. IEEE, 2020, pp. 237–242, DOI: 10.1109/iciot48696.2020.9089484.

[50] M. Hashemi, A. Mirrashid, and A. Beheshti Shirazi, "Driver safety development: Real-time driver drowsiness detection system based on convolutional neural network," *SN Computer Science*, vol. 1, no. 5, pp. 1–10, 2020, DOI: 10.1007/s42979-020-00306-9.

[51] A. Bagaskara and M. Suryanegara, "Evaluation of VGG-16 and VGG-19 deep learning architecture for classifying dementia people," in *2021 4th International Conference of Computer and Informatics Engineering (IC2IE)*. IEEE, sep 2021, DOI: 10.1109/ic2ie53219.2021.9649132.

[52] J.-M. Guo and H. Markoni, "Driver drowsiness detection using hybrid convolutional neural network and long short-term memory," *Multimedia tools and applications*, vol. 78, no. 20, pp. 29059–29087, 2019, DOI: 10.1007/s11042-018-6378-6.

[53] M. Ngxande, J.-R. Tapamo, and M. Burke, "Driver drowsiness detection using behavioral measures and machine learning techniques: A review of state-of-art techniques," *2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)*, pp. 156–161, 2017, DOI: 10.1109/robomech.2017.8261140.

[54] M. Remeli, S. Lestyán, G. Acs, and G. Biczók, "Automatic driver identification from in-vehicle network logs," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 1150–1157, DOI: 10.1109/itsc.2019.8917354.

[55] J. Yang, R. Zhao, M. Zhu, D. Hallac, J. Sodnik, and J. Leskovec, "Driver2vec: Driver identification from automotive data," *arXiv preprint arXiv:2102.05234*, 2021.

[56] D. Hallac, A. Sharang, R. Stahlmann, A. Lamprecht, M. Huber, M. Roehder, J. Leskovec et al., "Driver identification using automobile sensor data from a single turn," in *2016 IEEE 19th International ConferenceonIntelligentTransportationSystems (ITSC)*. IEEE, 2016, pp. 953–958, DOI: 10.1109/itsc.2016.7795670.

[57] N. Lin, C. Zong, M. Tomizuka, P. Song, Z. Zhang, and G. Li, "An overview on study of identification of driver behavior characteristics for automotive control," *Mathematical Problems in Engineering*, vol. 2014, 2014, DOI: 10.1155/2014/569109.

[58] G. A. M. Meiring and H. C. Myburgh, "A review of intelligent driving style analysis systems and related artificial intelligence algorithms," *Sensors*, vol. 15, no. 12, pp. 30 653–30 682, 2015, DOI: 10.3390/s151229822.

[59] M. Shahverdy, M. Fathy, R. Berangi, and M. Sabokrou, "Driver behavior detection and classification using deep convolutional neural networks," *Expert Systems with Applications*, vol. 149, p. 113240, 2020, DOI: 10.1016/j.eswa.2020.113240.

[60] J. Ferreira, E. Carvalho, B. V. Ferreira, C. de Souza, Y. Suhara, A. Pentland, and G. Pessin, "Driver behavior profiling: An investigation with different smartphone sensors and machine learning," *PLoS one*, vol. 12, no. 4, p. e0174959, 2017, DOI: 10.1371/journal.pone.0174959.

[61] Z. Marafie, K.-J. Lin, D. Wang, H. Lyu, Y. Meng, and T. Ito, "Autocoach: Driving behavior management using intelligent iot services," in *2019 IEEE 12th Conference on Service-Oriented Computing and Applications (SOCA)*. IEEE, 2019, pp. 103–110, DOI: 10.1109/soca.2019.00023.

[62] J.-L. Yin and B.-H. Chen, "An advanced driver risk measurement system for usage-based insurance on big driving data," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 4, pp. 585–594, 2018, DOI: 10.1109/tiv.2018.2874530.

[63] Y. Bian, C. Yang, J. L. Zhao, and L. Liang, "Good drivers pay less: A study of usage-based vehicle insurance models," *Transportation research part A: policy and practice*, vol. 107, pp. 20–34, 2018, DOI: 10.1016/j.tra.2017.10.018.

[64] M. F. Carfora, F. Martinelli, F. Mercaldo, V. Nardone, A. Orlando, A. Santone, and G. Vaglini, "A "pay-how-you-drive" car insurance approach through cluster analysis," *Soft Computing*, vol. 23, no. 9, pp. 2863–2875, 2019, DOI: 10.1007/s00500-018-3274-y.

[65] A. F. Abbas, U. U. Sheikh, F. T. AL-Dhief, and M. N. H. Mohd, "A comprehensive review of vehicle detection using computer vision," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 19, no. 3, pp. 838–850, 2021, DOI: 10.12928/telkomnika.v19i3.12880.

[66] A. Bouguettaya, H. Zarzour, A. Kechida, and A. M. Taberkit, "Vehicle detection from uav imagery with deep learning: a review," *IEEE Transactions on Neural Networks and Learning Systems*, 2021, DOI: 10.1109/tnnls.2021.3080276.

[67] J. Tang, L. Wan, J. Schooling, P. Zhao, J. Chen, and S. Wei, "Automatic number plate recognition (anpr) in smart cities: A systematic review on technological advancements and application cases," *Cities*, vol. 129, p. 103833, 2022, DOI: 10.1016/j.cities.2022.103833.

[68] M. Dasgupta, O. Bandyopadhyay, and S. Chatterji, "Automated helmet detection for multiple motorcycle riders using cnn," in *2019 IEEE Conference on Information and Communication Technology*. IEEE, 2019, pp. 1–4, DOI: 10.1109/cict48419.2019.9066191.

[69] J. B. Cicchino, "Effectiveness of forward collision warning and autonomous emergency braking systems in reducing front-to-rear crash rates," *Accident Analysis & Prevention*, vol. 99, pp. 142–152, 2017, DOI: 10.1016/j.aap.2016.11.009.

[70] M. H. Tawfeek and K. El-Basyouny, "A perceptual forward collision warning model using naturalistic driving data," *Canadian Journal of Civil Engineering*, vol. 45, no. 10, pp. 899–907, 2018, DOI: 10.1139/cjce-2017-0592.

[71] W. Song, Y. Yang, M. Fu, Y. Li, and M. Wang, "Lane detection and classification for forward collision warning system based on stereo vision," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5151–5163, 2018, DOI: 10.1109/jsen.2018.2832291.

[72] J. Tang, S. Li, and P. Liu, "A review of lane detection methods based on deep learning," *Pattern Recognition*, vol. 111, p. 107623, 2021, DOI: 10.1016/j.patcog.2020.107623.

[73] Q. Zou, H. Jiang, Q. Dai, Y. Yue, L. Chen, and Q. Wang, "Robust lane detection from continuous driving scenes using deep neural networks," *IEEE transactions on vehicular technology*, vol. 69, no. 1, pp. 41–54, 2019, DOI: 10.1109/tvt.2019.2949603.

[74] A. De la Escalera, J. M. Armingol, and M. Mata, "Traffic sign recognition and analysis for intelligent vehicles," *Image and vision computing*, vol. 21, no. 3, pp. 247–258, 2003.

[75] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition," *Neural networks*, vol. 32, pp. 323–332, 2012, DOI: 10.1016/j.neunet.2012.02.016.

[76] S. P. Rajendran, L. Shine, R. Pradeep, and S. Vijayaraghavan, "Real-time traffic sign recognition using yolov3 based detector," in *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE, 2019, pp. 1–7, DOI: 10.1109/icccnt45670.2019.8944890.

[77] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The german traffic sign recognition benchmark: a multi-class classification competition," in *The 2011 international joint conference on neural networks*. IEEE, 2011, pp. 1453–1460.

[78] J. Li and Z. Wang, "Real-time traffic sign recognition based on efficient cnns in the wild," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 975–984, 2018, DOI: 10.1109/tits.2018.2843815.

[79] S. Ftaimi and T. Mazri, "A comparative study of machine learning algorithms for vanet networks," in *Proceedings of the 3rd International Conference on Networking, Information Systems & Security*, 2020, pp. 1–8, DOI: 10.1145/3386723.3387829.

[80] F. Tang, B. Mao, N. Kato, and G. Gui, "Comprehensive survey on machine learning in vehicular network: technology, applications and challenges," *IEEE Communications Surveys & Tutorials*, 2021, DOI: 10.1109/comst.2021.3089688.

[81] M. N. Mejri, J. Ben-Othman, and M. Hamdi, "Survey on vanet security challenges and possible cryptographic solutions," *Vehicular Communications*, vol. 1, no. 2, pp. 53–66, 2014, DOI: 10.1016/j.vehcom.2014.05.001.

[82] A. M. Alrehan and F. A. Alhaidari, "Machine learning techniques to detect ddos attacks on vanet system: a survey," in *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*. IEEE, 2019, pp. 1–6, DOI: 10.1109/cais.2019.8769454.

[83] N. Kadam and R. S. Krovi, "Machine learning approach of hybrid ksvn algorithm to detect ddos attack in vanet," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 7, 2021, DOI: 10.14569/ijacsa.2021.0120782.

[84] H. Polat, M. Turkoglu, and O. Polat, "Deep network approach with stacked sparse autoencoders in detection of ddos attacks on sdn-based vanet," *IET Communications*, vol. 14, no. 22, pp. 4089–4100, 2020, DOI: 10.1049/iet-com.2020.0477.

[85] A. Acharya and J. Oluoch, "A dual approach for preventing blackhole attacks in vehicular ad hoc networks using statistical techniques and supervised machine learning," in *2021 IEEE International Conference on Electro Information Technology (EIT)*. IEEE, 2021, pp. 230–235, DOI: 10.1109/eit51626.2021.9491885.

[86] S. Pandey and V. Singh, "Blackhole attack detection using machine learning approach on manet," in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*. IEEE, 2020, pp. 797–802, DOI: 10.1109/icesc48915.2020.9155770.

[87] G. Farahani, "Black hole attack detection using k-nearest neighbor algorithm and reputation calculation in mobile ad hoc networks," *Security and Communication Networks*, vol. 2021, 2021.

[88] S. So, P. Sharma, and J. Petit, "Integrating plausibility checks and machine learning for misbehavior detection in vanet," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2018, pp. 564–571, DOI: 10.1109/icmla.2018.00091.

[89] R. W. van der Heijden, T. Lukaseder, and F. Kargl, "VeReMi: A dataset for comparable evaluation of misbehavior detection in VANETs," in *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*. Springer International Publishing, 2018, pp. 318–337, DOI: 10.1007/978-3-030-01701-9_18.

[90] A. Sharma and A. Jaekel, "Machine learning approach for detecting location spoofing in vanet," in *2021 International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2021, pp. 1–6, DOI: 10.1109/icccn52240.2021.9522170.

[91] B. Hammi, M. Y. Idir, and R. Khatoun, "A machine learning based approach for the detection of sybil attacks in c-its," in *2022 23rd Asia-Pacific Network Operations and Management Symposium (APNOMS)*. IEEE, 2022, pp. 1–4, DOI: 10.23919/apnoms56106.2022.9919991.

[92] J. Kamel, F. Haidar, I. B. Jemaa, A. Kaiser, B. Lonc, and P. Urien, "A misbehavior authority system for sybil attack detection in c-its," in *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*. IEEE, 2019, pp. 1117–1123, DOI: 10.1109/uemcon47517.2019.8993045.

[93] R. Rahal, A. Amara Korba, N. Ghoualmi-Zine, Y. Challal, and M. Y. Ghamri-Doudane, "Antibotv: A multilevel behaviour-based framework for botnets detection in vehicular networks," *Journal of Network and Systems Management*, vol. 30, no. 1, pp. 1–40, 2022, DOI: 10.1007/s10922-021-09630-8.

[94] Q. G. Fan, L. Wang, Y. N. Cai, Y. Q. Li, and J. Chen, "Vanet routing replay attack detection research based on svm," in *MATEC Web of Conferences*, vol. 63. EDP Sciences, 2016, p. 05020, DOI: 10.1051/matecconf/20166305020.

[95] H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep reinforcement learning based resource allocation for v2v communications," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3163–3173, 2019, DOI: 10.1109/icc.2018.8422586.

[96] J. Gao, M. R. Khandaker, F. Tariq, K.-K. Wong, and R. T. Khan, "Deep neural network based resource allocation for v2x communications," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*. IEEE, 2019, pp. 1–5, DOI: 10.1109/vtcfall.2019.8891446.

[97] M. E. Morocho-Cayamcela, H. Lee, and W. Lim, "Machine learning to improve multi-hop searching and extended wireless reachability in v2x," *IEEE Communications Letters*, vol. 24, no. 7, pp. 1477–1481, 2020, DOI: 10.1109/lcomm.2020.2982887.

[98] D. C. Moreira, I. M. Guerreiro, W. Sun, C. C. Cavalcante, and D. A. Sousa, "Qos predictability in v2x communication with machine learning," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. IEEE, 2020, pp. 1–5, DOI: 10.1109/vtc2020-spring48590.2020.9129490.

[99] C. Mo, Y. Li, and L. Zheng, "Simulation and analysis on overtaking safety assistance system based on vehicle-to-vehicle communication," *Automotive Innovation*, vol. 1, no. 2, pp. 158–166, 2018, DOI: 10.1007/s42154-018-0017-9.

[100] A. Vinel, L. Lan, and N. Lyamin, "Vehicle-to-vehicle communication in c-acc/platooning scenarios," *IEEE Communications Magazine*, vol. 53, no. 8, pp. 192–197, 2015, DOI: 10.1109/mcom.2015.7180527.

[101] M. Strunz, J. Heinovski, and F. Dressler, "Coop: V2v-based cooperative overtaking for platoons on freeways," in *ITSC*, 2021, pp. 1090–1097, DOI: 10.1109/itsc48978.2021.9565122.

[102] R. Miucic, A. Sheikh, Z. Medenica, and R. Kunde, "V2x applications using collaborative perception," in *2018 IEEE 88th Vehicular Technology Conference (VTC-Fall)*. IEEE, 2018, pp. 1–6, DOI: 10.1109/vtcfall.2018.8690818.

[103] J.-H. Shin, H.-B. Jun, and J.-G. Kim, "Dynamic control of intelligent parking guidance using neural network predictive control," *Computers & Industrial Engineering*, vol. 120, pp. 15–30, 2018, DOI: 10.1016/j.cie.2018.04.023.

[104] J. Liu, J. Wu, and L. Sun, "Control method of urban intelligent parking guidance system based on internet of things," *Computer Communications*, vol. 153, pp. 279–285, 2020, DOI: 10.1016/j.comcom.2020.01.063.

[105] D. B. Priya, V. R. Rao, and C. V. Kumar, "Intelligent parking system," *Int. Journal of Recent Technology and Engineering*, no. Part V, pp. 70–83, 2019.

[106] W. Yang and P. T. Lam, "Evaluation of drivers' benefits accruing from an intelligent parking information system," *Journal of cleaner production*, vol. 231, pp. 783–793, 2019, DOI: 10.1016/j.jclepro.2019.05.247.

[107] L. Jian, Z. Li, X. Yang, W. Wu, A. Ahmad, and G. Jeon, "Combining unmanned aerial vehicles with artificial-intelligence technology for traffic-congestion recognition: Electronic eyes in the skies to spot clogged roads," *IEEE Consumer Electronics Magazine*, vol. 8, no. 3, pp. 81–86, may 2019, DOI: 10.1109/mce.2019.2892286.

[108] I. Bisio, C. Garibotto, H. Haleem, F. Lavagetto, and A. Sciarrone, "A systematic review of drone based road traffic monitoring system," *IEEE Access*, vol. 10, pp. 101 537–101 555, 2022, DOI: 10.1109/access.2022.3207282.

[109] X. Liu, W. Gao, D. Feng, and X. Gao, "Abnormal traffic congestion recognition based on video analysis," in *2020 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, aug 2020, DOI: 10.1109/mipr49039.2020.00016.

[110] R. C. Gatto and C. H. Q. Forster, "Audio-based machine learning model for traffic congestion detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 7200–7207, nov 2021, DOI: 10.1109/tits.2020.3003111.

[111] C. K. Leung, J. D. Elias, S. M. Minuk, A. R. R. de Jesus, and A. Cuzzocrea, "An innovative fuzzy logic-based machine learning algorithm for supporting predictive analytics on big transportation data," in *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, 2020, pp. 1–8, DOI: 10.1109/fuzz48607.2020.9177823.

[112] J. Wu, L. Zhou, C. Cai, F. Dong, J. Shen, and G. Sun, "Towards a general prediction system for the primary delay in urban railways," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 3482–3487, DOI: 10.1109/itsc.2019.8916868.

[113] B. Yu, Z. Guo, S. Asian, H. Wang, and G. Chen, "Flight delay prediction for commercial air transport: A deep learning approach," *Transportation Research Part E: Logistics and Transportation Review,* vol. 125, pp. 203–221, 2019, DOI: 10.1016/j.tre.2019.03.013.

[114] N. Chakrabarty, T. Kundu, S. Dandapat, A. Sarkar, and D. K. Kole, "Flight arrival delay prediction using gradient boosting classifier," in *Emerging technologies in data mining and information security.* Springer, 2019, pp. 651–659, DOI: 10.1007/978-981-13-1498-8_57.

[115] A. M. Nagy and V. Simon, "Survey on traffic prediction in smart cities," *Pervasive and Mobile Computing*, vol. 50, pp. 148–163, 2018, DOI: 10.1016/j.pmcj.2018.07.004.

[116] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-gcn: A temporal graph convolutional network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 9, pp. 3848–3858, 2019, DOI: 10.1109/tits.2019.2935152.

[117] K. Guo, Y. Hu, Z. Qian, H. Liu, K. Zhang, Y. Sun, J. Gao, and B. Yin, "Optimized graph convolution recurrent neural network for traffic prediction," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 2, pp. 1138–1149, 2020, DOI: 10.1109/tits.2019.2963722.

[118] D. A. Tedjopurnomo, Z. Bao, B. Zheng, F. Choudhury, and A. K. Qin, "A survey on modern deep neural network for traffic prediction: Trends, methods and challenges," *IEEE Transactions on Knowledge and Data Engineering*, 2020, DOI: 10.1109/tkde.2020.3001195.

[119] L. Zhao, J. Wang, J. Liu, and N. Kato, "Routing for crowd management in smart cities: A deep reinforcement learning perspective," *IEEE Communications Magazine*, vol. 57, no. 4, pp. 88–93, 2019, DOI: 10.1109/mcom.2019.1800603.

[120] J. Tassone and S. Choudhury, "A comprehensive survey on the ambulance routing and location problems," *arXiv preprint arXiv:2001.05288*, 2020.

[121] R. Bai, X. Chen, Z.-L. Chen, T. Cui, S. Gong, W. He, X. Jiang, H. Jin, J. Jin, G. Kendall et al., "Analytics and machine learning in vehicle routing research," *International Journal of Production Research*, pp. 1–27, 2021, DOI: 10.1080/00207543.2021.2013566.

[122] T. D. H. Hussein, M. Frikha, S. Ahmed, and J. Rahebi, "Ambulance vehicle routing in smart cities using artificial neural network," in *2022 6th International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*. IEEE, 2022, pp. 1–6, DOI: 10.1109/atsip55956.2022.9805857.

[123] T. Darwassh Hanawy Hussein, M. Frikha, S. Ahmed, and J. Rahebi, "Ba-cnn: Bat algorithm-based convolutional neural network algorithm for ambulance vehicle routing in smart cities," *Mobile Information Systems*, vol. 2022, 2022, DOI: 10.1155/2022/7339647.

[124] X.-S. Yang and A. H. Gandomi, "Bat algorithm: a novel approach for global engineering optimization," *Engineering computations*, 2012, DOI: 10.1108/02644401211235834.

[125] D. Nallaperuma, R. Nawaratne, T. Bandaragoda, A. Adikari, S. Nguyen, T. Kempitiya, D. De Silva, D. Alahakoon, and D. Pothuhera, "Online incremental machine learning platform for big data-driven smart traffic management," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4679–4690, 2019, DOI: 10.1109/tits.2019.2924883.

[126] H. Wei, N. Xu, H. Zhang, G. Zheng, X. Zang, C. Chen, W. Zhang, Y. Zhu, K. Xu, and Z. Li, "Colight: Learning network-level cooperation for traffic signal control," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2019, pp. 1913–1922.

[127] T. Chu, J. Wang, L. Codecà, and Z. Li, "Multi-agent deep reinforcement learning for large-scale traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1086–1095, 2019, DOI: 10.1109/tits.2019.2901791.

[128] M. Guo, P. Wang, C.-Y. Chan, and S. Askary, "A reinforcement learning approach for intelligent traffic signal control at urban intersections," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2019, pp. 4242–4247, DOI: 10.1109/itsc.2019.8917268.

**Norman Bereczki** received his M.Sc. degree from the Budapest University of Technology and Economics (BME) in 2023. During his B.Sc. and M.Sc., he was part of the PARIPA university and industry cooperation programme and now is an active organizer of the PARIPA Alumni. He is currently doing his Ph.D. at the Department of Networked Systems and Services (HIT) in the Multimedia Networks and Services Laboratory (MEDIANETS). His research interests cover artificial intelligence use-cases for optimizing smart city services. He is currently working as a data scientist at Nitrowise Labs.

**Vilmos Simon** received his PhD from the Budapest University of Technology and Economics (BME) in 2009. Currently he is an Associate Professor at the Department of Networked Systems and Services and Head of the Multimedia Networks and Services Laboratory. He has done research on mobility management, energy efficiency in mobile cellular systems and self-organized mobile networks, recently his research interests include machine learning and data analytics for smart cities and intelligent transportation management systems. He published 60+ papers in international journals and conferences, and acts as a reviewer or organizer for numerous scientific conferences. He leaded the development of the first ITS system in Hungary.

# Closed-loop Orchestration for Cloud-native Mobile IPv6

Ákos Leiter[1], Edina Lami[1,2], Attila Hegyi[1], József Varga[1], and László Bokor[2,3]

*Abstract*—**With the advent of Network Function Virtualization (NFV) and Software-Defined Networking (SDN), every network service type faces significant challenges induced by novel requirements. Mobile IPv6, the well-known IETF standard for network-level mobility management, is not an exemption. Cloud-native Mobile IPv6 has acquired several new capabilities due to the technological advancements of NFV/SDN evolution. This paper presents how automatic failover and scaling can be envisioned in the context of cloud-native Mobile IPv6 with closed-loop orchestration on the top of the Open Network Automation Platform. Numerical results are also presented to indicate the usefulness of the new operational features (failover, scaling) driven by the cloud-native approach and highlight the advantages of network automation in virtualized and softwarized environments.**

*Index Terms*—**IP mobility, CN-MIPv6, ONAP, failover, scaling**

## I. Introduction

Network Function Virtualization (NFV) and Software Defined Networking (SDN) have not left any Network Function (NF) untouched. Meanwhile, cloud systems, either virtual machine or container-based, have created new execution environments. Nowadays, cloud-native service provisioning can bring in failover and scaling scenarios more straightforwardly than ever before. This implicitly indicates the usage of orchestration, which helps to organize the right amount of resources to the right place in time. In this paper, we use Open Network Automation Platform (ONAP) [1] for the practical experiments. With ONAP, we can run automatic failover and scaling scenarios when specific circumstances are met. In our experiments, we entirely rely on the closed-loop orchestration platform of ONAP. This is where Mobile IPv6 (MIPv6) [2] comes into the picture, whose functionality and operational procedures can be extended using the cloud. MIPv6 is part of a broader protocol family called IP-level mobility management. The current technological trends of cloudification spanned over the NFV, and SDN paradigms, together with orchestration requirements, do not leave MIPv6-based mobility support untouched. Cloud-native Mobile IPv6 (CN-MIPv6) [3] [4]was proposed to meet the expectations and apply the benefits of the trends mentioned above.

This paper aims to show how failover and scaling can be applied to CN-MIPv6 in the context of closed-loop orchestration. Here we define failover as the time of restoring redundancy. Furthermore, the article presents numerical results and calculations on the utilization of failover, scaling, and availability in the case of cloud-native IP-level mobility management. At the end of the paper, an analysis shows the benefits of network automation from the reliability and redundancy point of view.

The remaining sections are organized as follows. Section II presents related works. The connection between ONAP and CN-MIPv6 is elaborated in Section III, followed by the measurements and numerical calculations in Section IV. Conclusions and possible future research directions are in Section V and Section VI, respectively.

## II. Related Works

### A. Literature

Cloudification of 5G network functions has been a trending paradigm. *Du et al.* [5] consider the cloud-native bases of 5G Access and Mobility Management Function (AMF).

Another member of IP-based mobility management, such as Proxy Mobile IPv6 [6], has also been shaped to be cloud-native: cloud-native Proxy Mobile IPv6 (CN-PMIPv6) [7]. Flow Mobility, a concept standardized for Proxy Mobile IPv6 (PMIPv6) and discussed in Section III for MIPv6, makes it possible to separate IP flows by 5-tuples and assign individual mobility policies for each flow.

In our work, the Single-Point-of-Failure (SPOF) problem is solved by immediate and automatic redeployment of the mobility anchor (*Home Agent*). Furthermore, the number of anchors can be increased dynamically with automatic scaling. Obviously, there are other strategies for mitigating the SPOF problem in IPv6-based mobility management. With the help of SDN and Openflow, IPv6-based mobility management can be implemented in many different ways [8] [9] [10] [11] [12].

*Dimitris Giatsios et al.* [13] examine the failover of Network Slices. The paper written by *Veronica Quintuna Rodriguez et al.* examines ONAP-based deployment and management of Network Slices [14] [15]. ONAP has also been used to enhance access discovery and selection functions in the 5G core network in an article by *Rahul Banerji et al.* [16].

Predictive failover of Virtual Network Functions (VNF) in the context of edge computing is presented by *Huawei Huang et al.* [17]. A unique programming language is shown to ensure fault tolerance in SDN by *Reitblatt et al.* [18]. Scaling and failover can also be executed with Kubernetes on the Pod level by Kubernetes Horizontal Pod Autoscaler (HPA) [19] and Kubernetes Deployment [20]. But these are not suitable for inter-Kubernetes cluster failover and scaling: ONAP can manage many Kubernetes clusters simultaneously. As our concept opens the way to machine learning applications of

[1] Nokia Bell Labs, Budapest, Hungary, (E-mail: {akos.leiter, attila.hegyi, jozsef.varga}@nokia-bell-labs.com; edina.lami@nokia.com)

[2] Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Technology and Economics, Budapest, Hungary, (E-mail: bokorl@hit.bme.hu)

[3] ELKH-BME Cloud Applications Research Group, BME Informatics Building, Budapest, Hungary

DOI: 10.36244/ICJ.2023.1.5

IPv6-based mobility management, it is worth mentioning that mobility management itself is also examined in the field of Machine Learning optimization [21] [22] [23].

There is a good collection of mobility management issues in 5G networks with low-latency services by *Johanna Heinonen et al.* [24]. The identified challenges include topology-aware gateway selection, handover management, and gateway relocation. Our proposal inherits features of automatic gateway relocation from the system architecture level: appropriate functions are triggered automatically in case of failures.

Network automation and mobility management are not only in the interest of scientific papers. There are several other forums and publications, like blogs, which are worth mentioning (e.g., [25] [26] [27]). To the best of our knowledge, no publicly available literature deals with integrating closed-loop orchestration, cloud-native IPv6-based mobility management, and ONAP. The whole operation model of CN-MIPv6 – presented in this paper – is a novel concept. Of course, the scope of our article is limited to failover and scaling scenarios of CN-MIPv6, but these are the essential use cases demonstrating the power of our scheme. The microservice architecture CN-MIPv6 used in this paper for evaluation can be found in [4].

### B. *Orchestration and ONAP in a nutshell*

Orchestration (in telecommunication) is about ensuring the right amount of resources at the right time and location for a particular service. In practice, this means enough pieces of virtual machines and containers must be placed behind a specific service.

Figure 1 depicts another function of orchestration which operates with control loops. The system watches the particular service's actual states continuously and consistently enforces the desired state. This is how closed-loop orchestration or Kubernetes Operators [28] work on a very high level.

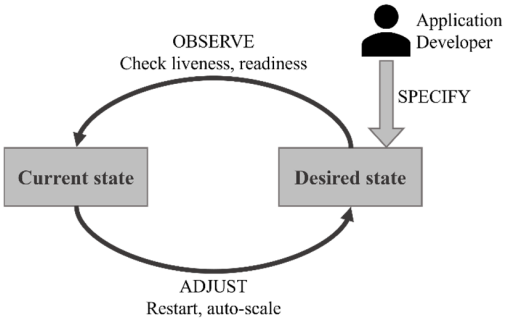



Figure 1 – Orchestration in general [29]

ONAP is an open-source network automation and closed-loop orchestration platform. A simplified view of its architecture can be seen in Figure 2. Its main component is the Service Orchestrator (SO) [30], which is responsible for executing abstract steps (Building Blocks, BB) to instantiate a particular Network Service (NS) instance. Network Service Models are designed in the Service Design and Creation (SDC) component [31]. These two components indicate that ONAP strictly separates the design time (Day 0) and the deployment time (Day 1). Particular workflows and controllers can be written to any NS with the help of the Controller Design Studio (CDS) [32]. CDS can also store Day 2 configuration changes or any other workflows intended to the part of a closed-loop control function. Software components of a Network Service Model, including Custom Controllers (Controller Blueprint Archive, CBA), are encapsulated as Vendor Software Package (VSP). In the case of CNF, a Helm chart is included in the VSP.

The Policy Framework [33] is the heart of the closed-loop functions of ONAP. It stores what to do when a particular event happens with an NF. The Policy Framework uses Policy Models, which hide the actual API calls to SO and CDS. Every Policy Model is assigned to a Network Service Model. The Onset event is sent out by a custom-made Analytic Application (AA) to trigger a Policy Model. Deciding when to send out an Onset message is entirely up to the developer of AA. The primary source of measurements for AA is VES messages sent out by a running NF. Every message, including Onset, *VNF Event Streaming (VES),* is transferred via DMAAP [34], which is the central message bus of ONAP.

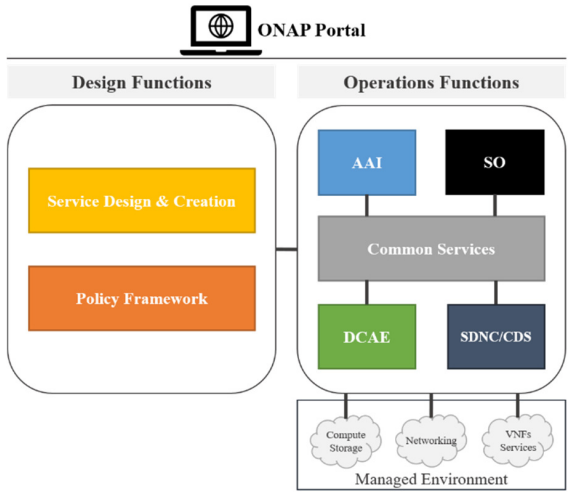



Figure 2 – The ONAP architecture [35]

## III. ONAP CONSIDERATIONS FROM THE CN-MIPV6 SERVICE PROVISIONING POINT OF VIEW

Our proposed approach of closed-loop orchestration in the context of CN-MIPv6 is depicted in Figure 3. On one hand, a Kubernetes layer is responsible for executing readiness and liveness probes for basic health checks. On the other hand, there is a broader scope of the control loop based on ONAP. This is where complex logic of automatic workflows is placed, such as analytics and machine learning-based operations. Of course, multiple control loops can be deployed here, such as autoscaling, auto-healing, etc. Furthermore, the scope of these control loops is not only restricted to one Kubernetes cluster. This is where inter-cloud control logic can be deployed.

In the context of ONAP and CN-MIPv6 integration, we use a multi-site VNF during the failover and scaling particularly. When a new instance of *Home Agent Packet Procesor* (*HA-PP)* is created, a new Virtual Function (VF) module is added to the existing VNF but with a different cloud region. This VF module hosts the corresponding Helm charts of *HA-PP*, which are added as a VSP to that particular Network Service Model.

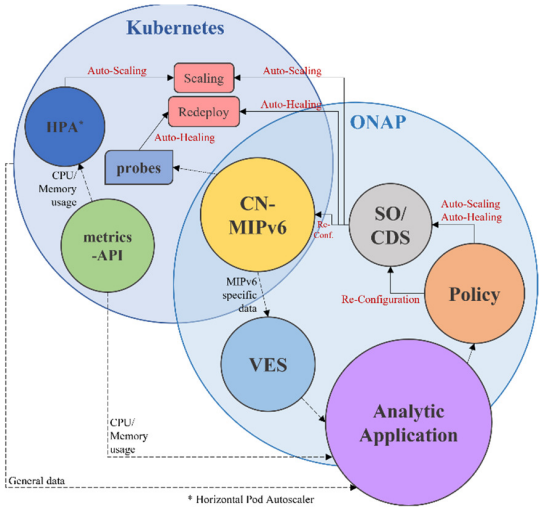



Figure 3 – Orchestration levels of CN-MIPv6

## IV. MEASUREMENTS AND NUMERICAL ANALYSIS

### A. *Testbed design*

The testbed runs on a Nokia Airframe server with Ubuntu 18.04 LTS. On the top of one VM, Kubernetes 1.20.2 is installed, which hosts the *HA-PPs*. From a simplicity point of view, we executed a namespace-to-namespace failover and scaling in our experiments. Of course, such use cases can also be applicable in a multi-cloud environment. Figure 4 shows the low-level design of the testbed from networking point-of-view. *Mobile Node (MN)* connects to a router (R) in order to add additional network between *MN* and *HA* (not being directly connected via home link). *Corresponding Node (CN)* represent a node which is not inside the mobility management domain.

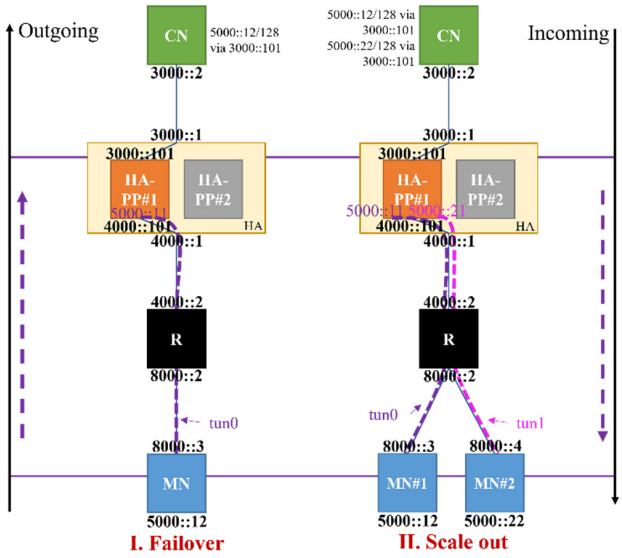



Figure 4 – Low-level design of the testbed

We added readiness and liveness probes to our *HA-PP* Pod. These probes monitor whether the corresponding RAW socket for sending and receiving is open. The following values are used:

- Initial delay of Readiness Probe: 20 sec
- PeriodSeconds of Readiness Probe: 90 sec
- Initial delay of Liveness Probe: 15 sec
- PeriodSeconds of Liveness Probe: 60 sec

### B. *Utilization for Failover*

Failover is the process when the traffic to a malfunctioning network function is offloaded to a working one. Our measurements logged how long it takes for an automatic failover managed by ONAP toolsets. In the context of CN-MIPv6, this means adding a new *HA-PP* to a different namespace, and at the end of the failover, this new *HA-PP* will be the new anchor point.

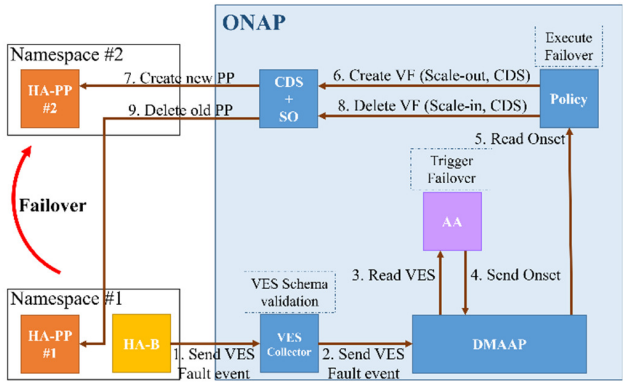



Figure 5 – ONAP-based execution workflow of the failover use case

Failover execution steps from the ONAP point of view are depicted in Figure 5 with the following explanation:

1. VES fault message is sent out from *Home Agent Backend (HA-B).*
2. VES collector catches VES fault message, does scheme validation, and sends it out to DMAAP.
3. AA reads the corresponding DMAAP topic continuously, whether or not a new message arrives. If yes, based on its own logic, it decides what Policy Model to trigger via the Onset message.
4. Onset message is sent out to the corresponding topic of DMAAP.
5. The Policy framework reads that topic in DMAAP.
6. The Policy calls the Scale-out workflow (Create VF module).
7. CDS and SO create the new VF module (Helm chart) instance in a different namespace.
8. The Policy executes Scale-in workflow in CDS.
9. CDS and SO delete the existing VF module (Helm chart) in the original namespace. The failover management procedure is finished.

Figure 6 shows the network traffic during failover. The content of the data plane is emulated by ICMPv6 messages. First, *MN* attaches to *HA-PP#1*. When *HA-PP#1* fails, a new *Binding Update (BU)* is sent out to *HA-PP#2* from *MN,* and the *IPv6inIPv6 tunnel* is set up to the new *HA-PP*. The *CN* is now reachable via *HA-PP#2*. *MN* can detect *HA* failure as routing is no longer working towards the tunnel. The details of the IPv6 address allocation during failover can be found at the end of the paper in Figure 12 and Figure 13. *Home Address* (*HoA)* is represented by the tunnel endpoint IPv6 address of MN, which is permanent. After the failover, the routing in connection with the *HA-PP* and *CN* is modified as the *MN,* and *HoA* is reachable via a different forwarding entity (different *HA-PP)*. Of course, this can be updated by any dynamic routing protocol or static routing, but this is not in the scope of this paper.

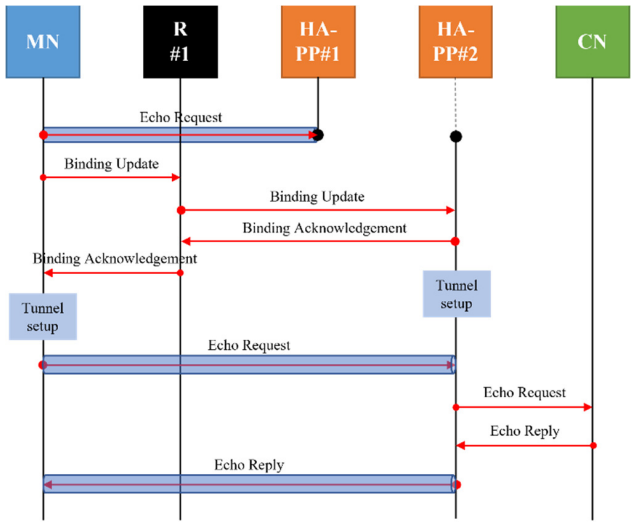



Figure 6 – CN-MIPv6 signaling flow in case of ONAP- based failover

A complete test framework is used with several measurement points for the process mentioned above. The exact measurement steps of failover can be seen at the end of the paper in Figure 14. The test framework is a separate entity measuring service outage time. By service outage, we mean the time when the bidirectional tunnel is broken and the new one is set up on a newly created *HA-PP*. Furthermore, the test framework connects to all the elements to initialize BU and adjust routing before and after failover, details in Figure 12, Figure 13, Figure 14, Figure 15, Figure 16 and Figure 17.

We have executed the failover scenario 100 times. The corresponding box plot can be seen in Figure 7. Meanwhile, numerical results are shown in Table 1. The failover is executed within 106.26 sec on average (median: 103.93, stdev: 30.42. The maximum is 210.19 sec, while the minimum is 58.83 sec. During measurements, we experienced that, sometimes, ONAP waited for an uncertain time during the same process. This explains the stdev and the high difference between the minimum and maximum.

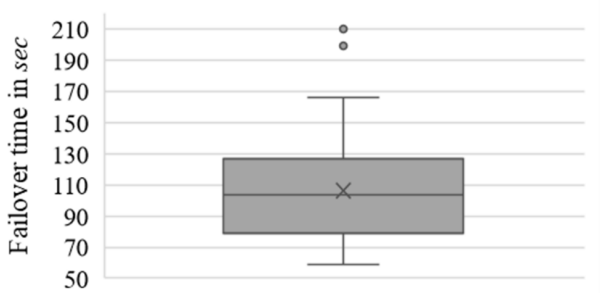



Figure 7 – Failover time results

TABLE I
NUMERICAL RESULTS OF FAILOVER TIME (SEC)

| MIN | AVG | Median | MAX | STDEV |
|---|---|---|---|---|
| 58.83 | 106.26 | 103.92 | 210.19 | 30.42 |

### C. *Utilisation for Scaling*

Scaling is the workflow when additional executors are added or removed to/from the system in order to deal with the changed traffic. In the case of our CN-MIPv6 mobility management service, we emulated how to add a new *HA-PP* when a certain traffic threshold is reached. At the end of the automatic scaling, both *HA-PP* components will serve the traffic.

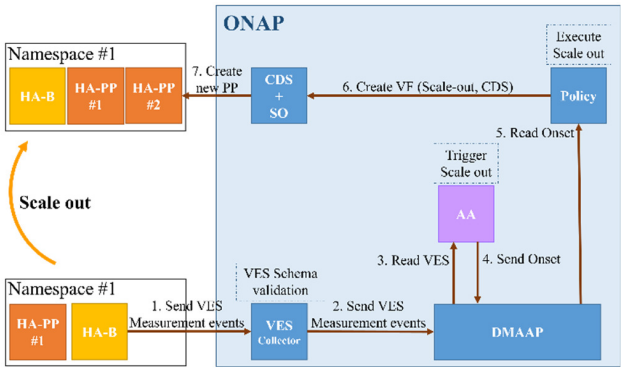



Figure 8 – ONAP-based execution workflow of scaling

Scaling execution steps from the ONAP point of view - shown in Figure 8 – are the followings:

1. *HA-B* sends out VES Measurement event.
2. VES Collector validates its schema and puts it to the corresponding DMAAP topic.
3. AA gets the VES message and calculates if scaling is needed or not.
4. AA sends out the Onset message to DMAAP to trigger a Policy model.
5. The Policy gets the Onset message.
6. The Policy calls for scale-out workflow in CDS.
7. CDS and SO executes the scale-out request and create a new VF module instance in a different namespace.

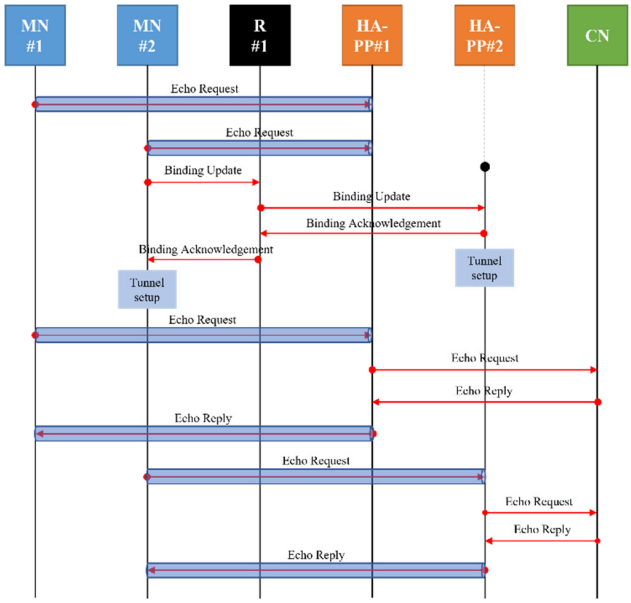



Figure 9 – Signaling of CN-MIPv6 in case of scaling

Figure 9 shows how scaling is executed from the CN-MIPv6 point of view. First, two *MNs (MN#1, MN#2)* are attached to one *HA-PP (HA-PP#1)*. After the ONAP recognizes the overload of *HA-PP#1*, then the management system creates a new one. From this point, *MN#2* is reregistered via sending BU to the newly created *HA-PP#2*, and the bidirectional tunnel is set up to *HA-PP#2* in the case of *MN#2*. Thus, the *CN* is reachable via *HA-PP#2* in the case of *MN#2*.

At the end of this paper, Figure 15 and Figure 17 present the IPv6 address allocation during scaling. Similar logic can be seen for IPv6 address allocation as in the case of failover.The detailed measurement steps of the scaling use case can be found at the end of the paper in Figure 16. This also uses the same elements as the failover scenario. Meanwhile, Pods' network interface capacities are limited to 1 Gbps to simulate the limited capacity of a network device and packet processing. The original throughput was about cc 900 Mpbs for the *MNs* using one *HA-PP* instance. This leads to a common base on measurement, and it is easier to compare results. The scaling scenario has been executed 20 times. Figure 10 and Figure 11 show the box plot of the results in Mbps and percentage, respectively. Table 2 shows the numerical results of the average gain for both *MNs*. After the scaling, the throughput gain for *MN#1* is 362.1 Mbps (67.94%), while on *MN#2,* it is 329.5Mbps (65.4%). *MN#2* shows a little less throughput gain, as it has an unconnected period which includes new binding message exchanges and tunnel setup time, while *MN#1* is continuously connected to the *HA-PP#1*.

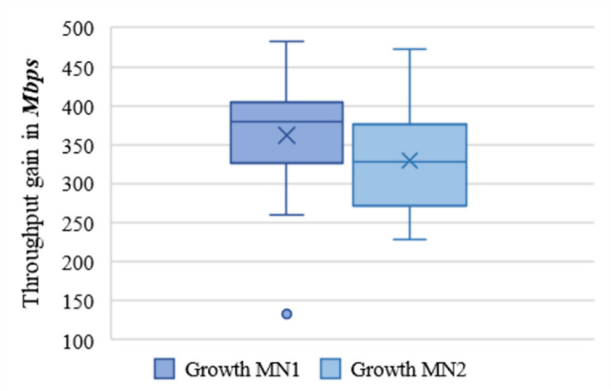



Figure 10 – Measured throughput gain after scaling out (in Mbps)

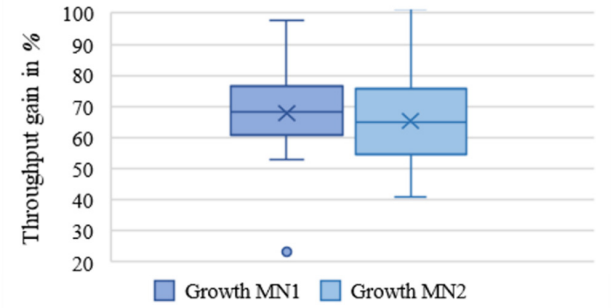



Figure 11 – Measured throughput gain after scaling out (in percentage)

TABLE II
NUMERICAL RESULTS OF AVERAGE THROUGHPUT GAIN AFTER SCALING

| MN1 throughput gain (Mbps) | MN1 throughput gain (%) | MN2 throughput gain (Mbps) | MN2 throughput gain (%) |
|---|---|---|---|
| 362.1 | 67.94 | 329.5 | 65.4 |

### *D. Availability and redundancy calculations*

The availability of a telecommunication system is one of the most critical quality metrics. By automating the failover procedure of a network function, the availability of the network function significantly improves.

A short recap, the availability of a single NF can be calculated by

$$A_s = \frac{MTBF}{\text{MTBF+MTTR}} \quad (1)$$

where MTBF is Mean Time Between Failures, MTTR is Mean Time to Repair.

Telco services provide high availability; for Ulra-Reliably Low-Latency (URLLC) services, these expectations are growing further. When the availability expectations for service are in the range of five or six nines for the service components, including the network function software, the availability expectation is even higher. These figures are achieved by applying redundancy schemes. There are no widely agreed/accepted figures for software MTBF and

MTTR values, so we calculated with a range of values (MTBF from a month to two years, MTTR from 5 minutes to 45 minutes). We applied different redundancy schemes for these values to ensure that the availability of the redundant solutions is above six nines. Obviously, the reduced MTTR in every case improves the availability. This improvement allows the deployment of lighter redundancy schemes for the network function when its failover is automated and the same availability is still provided. For example, a 2N redundant deployment allows deploying a 3+1 redundant scheme, and the same availability is provided (thus, for four software instances, three instances become active instead of two [36] [37]). Suppose the original MTTR was on the higher end (i.e., above 15 minutes). Even the 6+1 redundant scheme provides the same availability (for all MTBF values) as the 2N redundant solution without failover automation. Thus a significant amount of resources can be saved. Note that it is also possible to use the "saved availability budget" for other components of the system (e.g., lowering hardware availability by employing less personnel and saving cost) and keep end-to-end availability on the same level or simply offer better availability for customers.

With 2N and 3N redundant systems, the availability calculation is the following:

$$A_{2N} = 1 - (1 - A_S)^2 \quad (2)$$

$$A_{3N} = 1 - (1 - A_S)^3 \quad (3)$$

In Section IV. B, we have shown the average time for failover, which is 106.23 sec. This value can be considered as the MTTR of the function we evaluate.

The goal of these calculations is the following: in our ONAP-based failover case, it is possible to reduce the level of redundancy. This means the number of deployed instances can be decreased in order to save resources. Meanwhile, the availability of service is not jeopardized. For calculation simplicity reasons, we apply 99.999% availability for NF, but in a real-world scenario, higher availability is expected.

1N non-redundant case:

Based on the above-mentioned equations, we calculated the MTBF value for the 1N system if availability must be at least 99.999%, which is 123 days. This means the frequency of system collapse cannot be less than 123 days; otherwise, the system availability cannot reach 99.999%, supposed the cc.106 sec failover time

We also consider that this MTBF value describes the by-default behavior of our software system.

2N redundant case:

With the MTTR=106.23 sec value and 2N redundancy, the MTBF value is 9.3 hours ($MTBF_{2N}$=9.3 hours) if minimum 99.999% availability is kept. Thus, if the $MTBF_{2N}$ is 9.3 hours or less, the system is below the target of 99.999% availability. So, if every 9.3 hours, there is an failure with the given MTTR value, the system will still operate on at least 99.999% availability.

3N redundant case:

With the MTTR=106.23 sec value and 3N redundancy, the MTBF value is 1.3 hours ($MTBF_{3N}$=1.3 hours) if minimum 99.999% availability is kept. Thus, if the $MTBF_{3N}$ is 1.3 hours or less, the system is below the target of 99.999% availability. So, if every 1.3 hours, there is an error with the given MTTR value, the system still operates on at least 99.999% availability.

We can save resources if the same availability can be kept with fewer redundant pairs. If we know that our system has an error on average every 123 days, then let's check the maximum MTTR value that is allowed with 2N redundancy to keep at least 99.999% availability. This is 9.365 hours ($MTTR_{2N}$=9.365 hours). This means that every 2N system with MTBF = 123 days can be reduced to our 1N system, where the MTTR is minimum 9.365 hours. Because this MTTR value reaches a certain level, where the availability of the 2N system does not keep 99.999% even though it may be better than the 1N, but the over-availability requirement is not fulfilled. But using 1N instead of 2N means a service outage as there is no active pair to maintain service. To circumvent this, a 3N redundancy system is needed. A similar logic is applied: if the MTBF = 123 days, then let's calculate the min MTTR when the 3N system does not add availability gain (not reaching 99.999%) compared to the 2N system. Every 3N system can be reduced to a 2N system where the MTTR is between 12 min and 65 hours. The whole calculation is represented in Figure 18 at the end of the paper.

## V. CONCLUSION

In this paper, we have presented that the redundancy restoration time (restoring a cold backup) can be decreased to the level of minutes with network automation and orchestration. Note that according to the GSMA [38], a new instance deployment of a Physical Network Function lays in the range of days. In this paper, we have shown that this can be decreased to the range of minutes. This definitely shows the power of network automation; meanwhile, cloud, virtualization, and containerization are utilized as well.

This failover time is also considered to restore the cold backup of a particular network function. We also believe that this does not only pertain to CN-MIPv6; general conclusions can be drawn for any NF.

Even though we have shown that the redeployment time has gone to the range of minutes, there are further possibilities for optimization: detailed measurements are needed to conclude the minimum value of liveness and readiness probe to minimize the failover or scaling time. Right now, they are arbitrary. A new measurement point can also be added to the test framework, which watches the ONAP internal states to identify the slowest part of the execution accurately. Failover time may be higher in a real-world scenario because our testbed does not deal with complex routing and configuration. This is similar to the scaling as well because the gain may be lower due to the cloud's actual computing and network load uncertainty. We have also shown that, with the help of the scaling procedure, significant throughput gain can be reached.

Improvement of failover time also has a positive impact on service availability. It saves resources for the automated NF function or any other part of the chain of processes contributing to service provision.

## VI. FUTURE WORK

Current Kubernetes network approaches are not "network-native". More and more novel Kubernetes networking approaches are emerging [39] [40], which are the natural evolution steps of the current version of our PoC implementation. For a simplified but more advanced use case, using an external load balancer instead of a direct Pod connection is also in the scope of future implementations.

ONAP has also paved the way for machine learning applications, which may not only deal with traffic level optimization in failover or scaling. Machine learning can add new perspectives to predictive mobility management and other potential application areas.

## REFERENCES

[1] 'Open Network Automation Platform (ONAP)'. https://www.onap.org/ (accessed Jul. 28, 2022).

[2] C. Perkins, D. Johnson, and J. Arkko, *Mobility Support in IPv6*. IETF, 2011. [Online]. Available: http://www.ietf.org/rfc/rfc6275.txt

[3] Á. Leiter, L. Bokor, and I. Kispál, 'An Evolution of Mobile IPv6 to the Cloud', in *Proceedings of the 18th ACM Symposium on Mobility Management and Wireless Access*, New York, NY, USA, 2020, pp. 137–141. DOI: 10.1145/3416012.3424633.

[4] Á. Leiter et al., 'Cloud-native IP-based mobility management: a MIPv6 Home Agent standalone microservice design', in *2022 13th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, 2022, pp. 252–257. DOI: 10.1109/CSNDSP54353.2022.9908059.

[5] K. Du, X. Wen, L. Wang, and T.-T. Nguyen, 'A Cloud-Native Based Access and Mobility Management Function Implementation in 5G Core', in *2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, 2020, pp. 1251–1256. DOI: 10.1109/ICCC51575.2020.9345262.

[6] S. Gundavelli (Ed.), K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil, *Proxy Mobile IPv6*. Fremont, CA, USA: RFC Editor, 2008. DOI: 10.17487/RFC5213.

[7] Á. Leiter, N. Galambosi, and L. Bokor, 'An Evolution of Proxy Mobile IPv6 to the Cloud', in *Proceedings of the 19th ACM International Symposium on Mobility Management and Wireless Access*, New York, NY, USA: Association for Computing Machinery, 2021, pp. 107–115. [Online]. Available: DOI: 10.1145/3479241.3486684

[8] S. Kim, H. Choi, P. Park, S. Min, and Y. Han, 'OpenFlow-based Proxy mobile IPv6 over software defined network (SDN)', in *2014 IEEE 11th Consumer Communications and Networking Conference (CCNC)*, Jan. 2014, pp. 119–125. DOI: 10.1109/CCNC.2014.6866558.

[9] S. M. Raza, D. S. Kim, D. Shin, and H. Choo, 'Leveraging proxy mobile IPv6 with SDN', *Journal of Communications and Networks*, vol. 18, no. 3, Art. no. 3, Jun. 2016, DOI: 10.1109/JCN.2016.000061.

[10] K. M. Sue, S. Kamolphiwong, T. Kamolphiwong, and L. Damyos, 'SDN Based Fast Handover over IP Mobility', in *2019 23rd International Computer Science and Engineering Conference (ICSEC)*, 2019, pp. 345–350. DOI: 10.1109/ICSEC47112.2019.8974787.

[11] K. Hee Lee, 'Mobility Management Framework in Software Defined Networks', *International Journal of Software Engineering and Its Applications*, vol. 8, no. 8, pp. 1–10.

[12] Á. Leiter, M. S. Saleh, L. Pap, and Bokor, 'Survey on PMIPv6-based Mobility Management Architectures for Software-Defined Networking', *Infocommunications Journal*, vol. XIV, no. 2, DOI: 10.36244/ICJ.2022.2.1.

[13] D. Giatsios, K. Choumas, P. Flegkas, T. Korakis, and D. Camps-Mur, 'SDN implementation of slicing and fast failover in 5G transport networks', in *2017 European Conference on Networks and Communications (EuCNC)*, 2017, pp. 1–6. DOI: 10.1109/EuCNC.2017.7980671.

[14] V. Q. Rodriguez, F. Guillemin, and A. Boubendir, 'Automating the deployment of 5G Network Slices using ONAP', in *2019 10th International Conference on Networks of the Future (NoF)*, 2019, pp. 32–39. DOI: 10.1109/NoF47743.2019.9015043.

[15] V. Q. Rodriguez, F. Guillemin, and A. Boubendir, '5G E2E Network Slicing Management with ONAP', in *2020 23rd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, 2020, pp. 87–94. DOI: 10.1109/ICIN48450.2020.9059507.

[16] R. Banerji et al., 'ONAP Based Pro-Active Access Discovery and Selection for 5G Networks', in *2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, 2020, pp. 1–6. DOI: 10.1109/WCNCW48565.2020.9124724.

[17] H. Huang and S. Guo, 'Proactive Failure Recovery for NFV in Distributed Edge Computing', *IEEE Communications Magazine*, vol. 57, no. 5, pp. 131–137, 2019, DOI: 10.1109/MCOM.2019.1701366.

[18] M. Reitblatt, M. Canini, A. Guha, and N. Foster, 'FatTire: Declarative Fault Tolerance for Software-Defined Networks', in *Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking*, New York, NY, USA, 2013, pp. 109–114. DOI: 10.1145/2491185.2491187.

[19] 'Kubernetes: Horizontal Pod Autoscaler'. https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/ (accessed Jul. 15, 2022).

[20] 'Deployment element of Kubernetes'. https://kubernetes.io/docs/concepts/workloads/controllers/deployment/ (accessed Mar. 12, 2021).

[21] J. W. Mwangoka, P. Marques, and J. Rodriguez, 'Cognitive Mobility Management in Heterogeneous Networks', in *Proceedings of the 8th ACM International Workshop on Mobility Management and Wireless Access*, New York, NY, USA, 2010, pp. 37–44. DOI: 10.1145/1868497.1868504.

[22] M. Simsek, M. Bennis, and I. Guvenc, 'Mobility management in HetNets: a learning-based perspective', *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, no. 1, p. 26, Feb. 2015, DOI: 10.1186/s13638-015-0244-2.

[23] R. Boutaba, N. Shahriar, M. A. Salahuddin, S. R. Chowdhury, N. Saha, and A. James, 'AI-Driven Closed-Loop Automation in 5G and beyond Mobile Networks', in *Proceedings of the 4th FlexNets Workshop on Flexible Networks Artificial Intelligence Supported Network Flexibility and Agility*, New York, NY, USA, 2021, pp. 1–6. DOI: 10.1145/3472735.3474458.

[24] J. Heinonen, P. Korja, T. Partti, H. Flinck, and P. Pöyhönen, 'Mobility management enhancements for 5G low latency services', in *2016 IEEE International Conference on Communications Workshops (ICC)*, 2016, pp. 68–73. DOI: 10.1109/ICCW.2016.7503766.

[25] Chantel Soumis, 'AMM: WHA T IS AUTOMA TED MANAGEMENT?' https://www.valicomcorp.com/blog/2018/4/2/amm-what-is-automated-mobility-management (accessed Mar. 21, 2022).

[26] Pallavi Vanacharla, 'SP360: Service Provider Winning business customers with automated mobility management'. https://blogs.cisco.com/sp/winning-business-customers-with-automated-mobility-management (accessed Mar. 21, 2022).

[27] Pallavi Vanacharla, 'Digital Transformation Automation: Moving beyond manual mobility management'. https://blogs.cisco.com/digital/automation-moving-beyond-manual-mobility-management (accessed Mar. 21, 2022).

[28] 'Kubernetes Operators'. https://kubernetes.io/docs/concepts/extend-kubernetes/operator/ (accessed Jan. 15, 2022).

[29] 'An introduction to closed-loop automation'. https://developer.ibm.com/articles/an-introduction-to-closed-loop-automation/ (accessed Mar. 21, 2022).

[30] 'ONAP – Service Orchestrator'. https://wiki.onap.org/pages/viewpage.action?pageId=1015834 (accessed Jul. 15, 2022).

[31] 'ONAP – Service Design and Creation'. https://wiki.onap.org/pages/viewpage.action?pageId=1015837 (accessed Jul. 15, 2022).

[32] 'ONAP – Controller Design Studio'. https://docs.onap.org/projects/onap-ccsdk-cds/en/latest/ (accessed Jul. 15, 2022).

[33] 'ONAP – Policy Framework'. https://docs.onap.org/projects/onap-policy-parent/en/latest/architecture/architecture.html (accessed Jul. 15, 2022).

[34] 'ONAP - Data Movement as a Platform'. https://wiki.onap.org/pages/viewpage.action?pageId=3247130 (accessed Jul. 15, 2022).

[35] F. Slim, F. Guillemin, A. Gravey, and Y. Hadjadj-Aoul, 'Towards a dynamic adaptive placement of virtual network functions under ONAP', in *2017 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, 2017, pp. 210–215. DOI: 10.1109/NFV-SDN.2017.8169880.

[36] A. Hilt, I. Bakos, and G. Járó, 'Reliability and availability modelling of telecommunication servers on cloud', in *2015 17th International Conference on Transparent Optical Networks (ICTON)*, 2015, pp. 1– 4. DOI: 10.1109/ICTON.2015.7193412.

[37] J. Varga, A. Hilt, J. Bíró, C. Rotter, and G. Jaro, 'Reducing operational costs of ultra-reliable low latency services in 5G', *Infocommunications Journal*, vol. X, pp. 37–45, 2018, DOI: 10.36244/ICJ.2018.4.6.

[38] GSMA, 'Migration from Physical to Virtual Network Functions: Best Practices and Lessons Learned'. https://www.gsma.com/futurenetworks/5g/migration-from-physical-to-virtual-network-functions-best-practices-and-lessons-learned/ (accessed Jul. 15, 2022).

[39] 'Kubernetes – Network Speciel Interest Group'. https://github.com/kubernetes/community/tree/master/sig-network (accessed Sep. 28, 2021).

[40] 'Network Service Mesh'. https://networkservicemesh.io/ (accessed Mar. 12, 2021).

**Ákos Leiter** has graduated as a Computer Engineer MSC in Department of Networked Systems and Services (HIT), Budapest University of Technology and Economics (BME) in 2015, specialized in Computer Networks. His theses was about proposing an operator-centric, dynamic flow mobility protocol with IP in Evolved Packet Core. Currently he is a PhD candidate at the Department of Networked Systems and Services in the Multimedia Networks and Services Laboratory (MEDIANETS) and a Research Engineer at Nokia Bell Labs. His main research field is Network Function Virtualization and Software Defined Networking including Orchestration and Network Automation. His work-in-progress Phd theses is about the cloudification of Mobile IPv6 protocol family on the top of Kubernetes.

**Edina Lami** is a solution expert at Nokia, technical lead of the Nokia Core Slicing team since 2022. She received her MSc degree in computer engineering from Budapest University of Technology and Economics in 2022. She wrote her thesis about the closed-loop orchestration of cloud-native IP-based mobility management. She joined Nokia Bell Labs in 2020 as a Research Engineer Trainee where she was involved in mobility management and orchestration related research topics.

**Attila Hegyi** received the MSc in computer science and mathematics from the University of Szeged (SZTE) in 2010. He had worked for multiple companies in the telecommunication domain as software engineer and currently he is a senior research engineer at Nokia Bell Labs. His main research topics are in the field of cloud-native network automation, multi-cloud orchestration and edge computing.

**József Varga**, PhD, is a senior research engineer at Nokia, member of the 'Multi Cloud Orchestration' research group in Nokia Bell Labs. He received his MSc in computer science and mathematics from the University of Szeged in 1991, PhD in IT from the University of Veszprém in 2002. He was an assistant professor at the University of Szeged, then at the University of Veszprém. He joined Nokia in 1999, he was involved in IP Multimedia Subsystem development, then represented Nokia as a standardization delegate in 3GPP from 2004 to 2011. In 2011 he joined Nokia Research Center (now Nokia Bell Labs) dealing with topics like SDN, virtualization, and orchestration. Currently he is focusing on resource management in 6G, including the economic aspects. He co-authored more than 10 papers and more than 10 granted patents.

**László Bokor** received his Ph.D. degree in computer engineering from Budapest University of Technology and Economics in 2014. He is currently an associate professor at the Dept. of Networked Systems and Services where he leads the Commsignia-BME HIT Automotive Communications Research Group. He is a member of the HTE, the Hungarian Standards Institution's Technical Committee for ITS (MSZT/MB 911), the TPEGoverC-ITS Task Force within the TPEG Application Working Group of TISA, the ITS Hungary Association, and the BME's MediaNets Laboratory. In recognition of his professional work and achievements in mobile telecommunications, he received the HTE Silver Medal (2013), the HTE Pollák-Virág Award (2015, 2022), and the HTE Gold Medal (2018). He was a recipient of the UNKP-16-4-I. Post-Doctoral Fellowship in 2016 from the New National Excellence Program of the Ministry of Human Capacities of Hungary. In 2018 he was awarded the Dean's Honor (BME VIK) for education and research achievements in the field of communication of autonomous vehicles; in 2020, he received the BME HIT Excellence in Education Award.

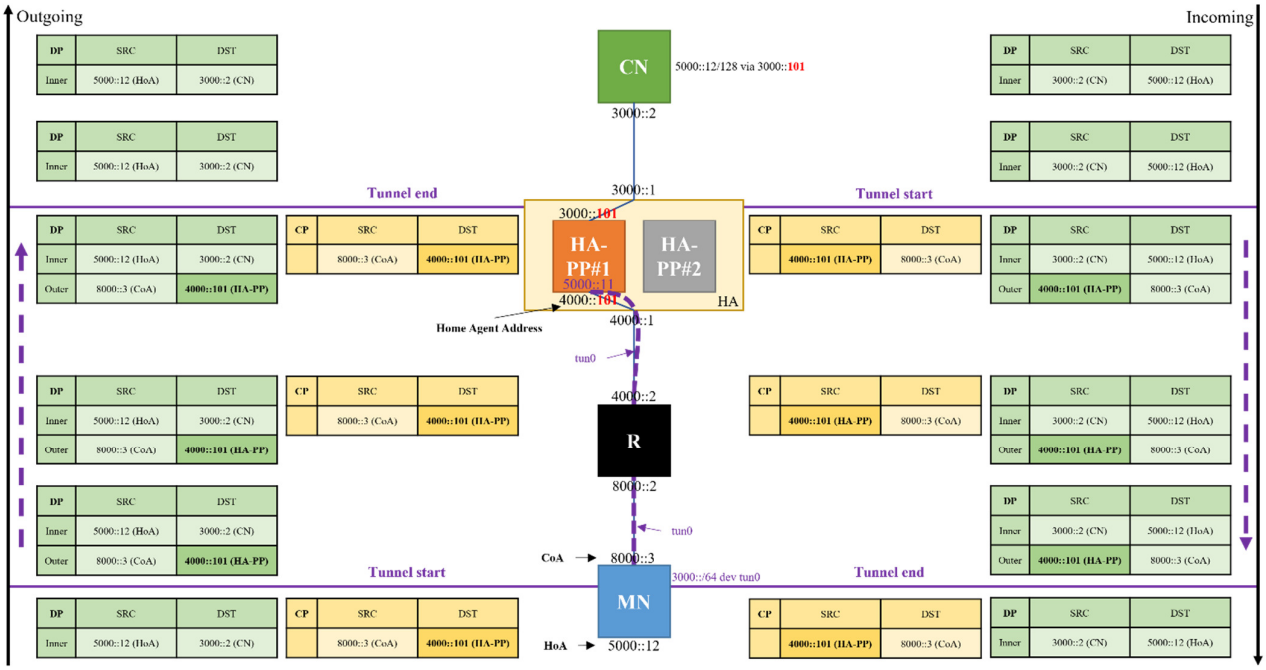



Figure 12 – IPv6 address allocation before failover

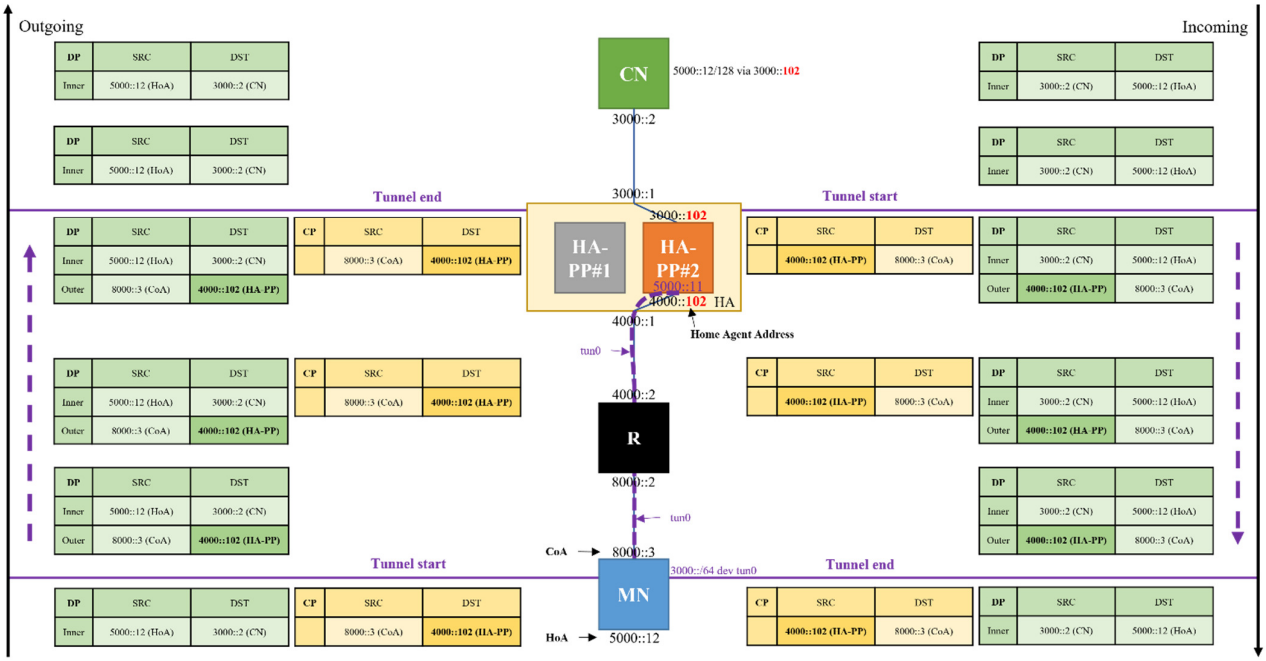



Figure 13 – IPv6 address allocation after failover

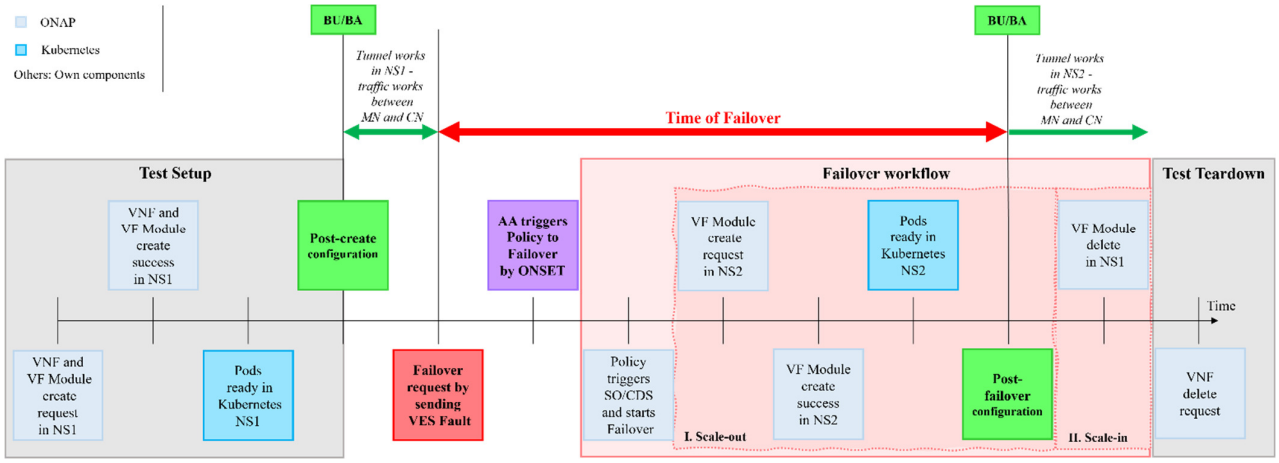



Figure 14 – Detailed description of the failover use case

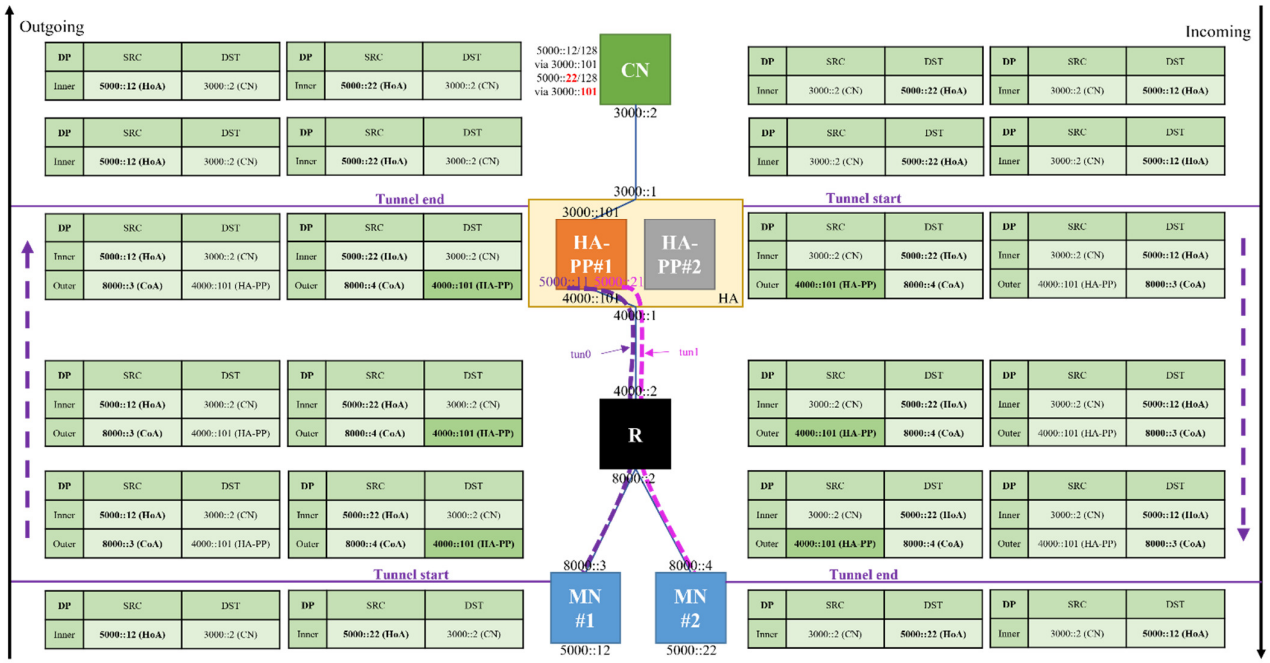



Figure 15 – IPv6 address allocation before scaling

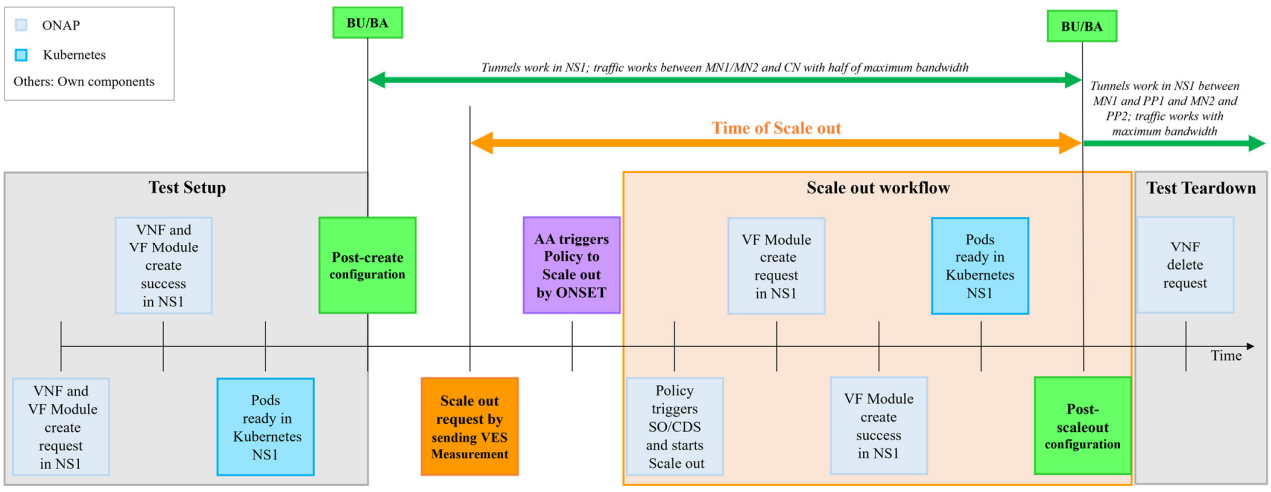



Figure 16 – Detailed description of the scaling use case

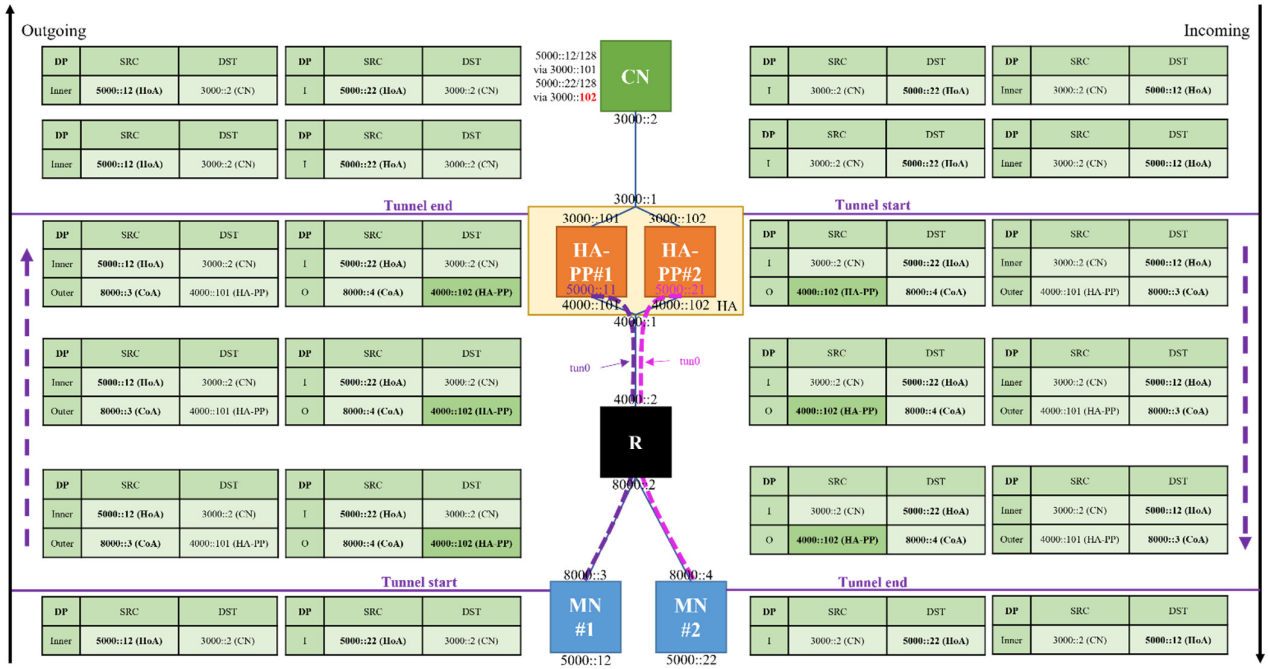



Figure 17 – IPv6 address allocation after scaling

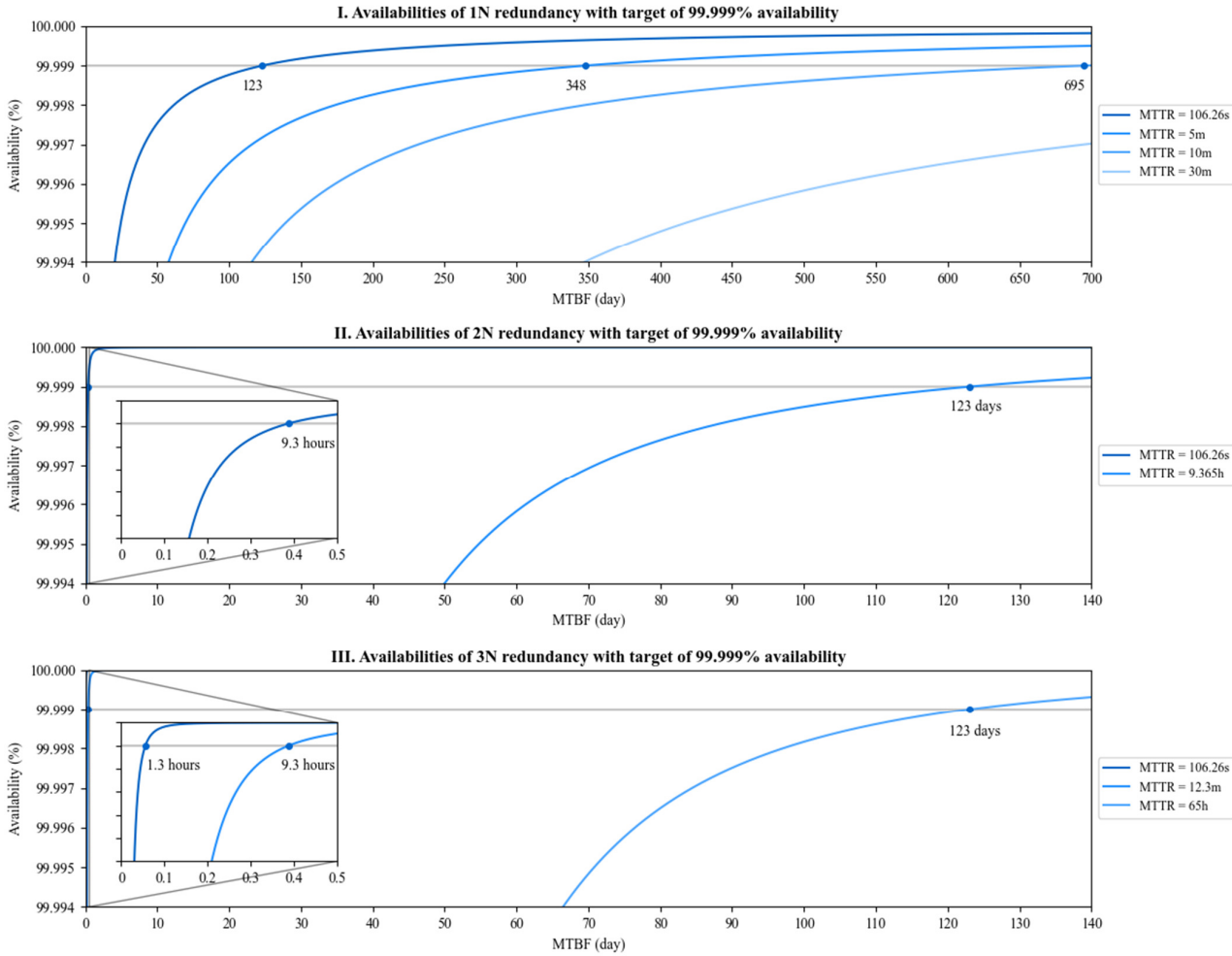



Figure 18 – Availability and redundancy calculations

# Optimizing the Performance of the Iptables Stateful NAT44 Solution

Gábor Lencse, Keiichi Shima

***Abstract*—The stateful NAT44 performance of iptables is an important issue when it is used as a stateful NAT44 gateway of a CGN (Carrier-Grade NAT) system. The performance measurements of iptables published in research papers do not comply with the requirements of RFC 2544 and RFC 4814 and the usability of their results has serious limitations. Our Internet Draft has proposed a benchmarking methodology for stateful NATxy (x, y are in {4, 6}) gateways and made it possible to perform the classic RFC 2544 measurement procedures like throughput, latency, frame loss rate, etc. with stateful NATxy gateways using RFC 4814 pseudorandom port numbers. It has also defined new performance metrics specific to stateful testing to quantify the connection setup and connection tear down performance of stateful NATxy gateways. In our current paper, we examine how the performance of iptables depends on various settings, and also if certain tradeoffs exist. We measure the maximum connection establishment rate, throughput and tear down rate of iptables as well as its memory consumption as a function of hash table size always using 40 million connections. We disclose all measurement details and results. We recommend new settings that enable network operators to achieve significantly higher performance than using the traditional ones.**

***Index Terms*—benchmarking, iptables, netfilter, optimization, performance, stateful NAT44.**

## I. INTRODUCTION

THE depletion of the public IPv4 address pool of IANA in 2011 has presented the ISPs (Internet Service Providers) with a dilemma: either they deploy IPv6 as soon as possible or they use CGN (Carrier-Grade NAT). We believe that the first one is the only workable solution in the long run, but we also experience that the transition to IPv6 is happening rather slowly for various reasons [1], and we estimate that IPv4 will be with us for decades. Therefore, stateful NAT44 (also called NAPT: Network Address and Port Translation) gateways will also be needed for a long time. The *Netfilter Framework* [2] of the Linux kernel (usually called *iptables* after the name of its command line management tool) is a widely used solution for this purpose.

Submitted: December 14, 2022.

This work was supported by the International Exchange Program of the National Institute of Information and Communications (NICT), Japan.

Gábor Lencse is with the Department of Telecommunications, Széchenyi István University, Győr, Hungary (e-mail: lencse@sze.hu).

Keiichi Shima is with the SoftBank Corporation, Minato-ku, Tokyo, Japan.

We are aware that in some areas of application, iptables is gradually replaced by *nftables*. The latter has advantages, when a high number of rules are used and they are often reconfigured, but it did not became an industry standard yet [3]. When implementing CGN, there is no need for a high number of rules and they are very rarely reconfigured, thus iptables is still appropriate for this purpose. What really matters for the ISPs, it is the performance of the stateful NAT44 translation. To that end, iptables is a good choice: the iptables stateful NAT44 solution outperformed the Jool NAT64 solution by an order of magnitude in throughput and its performance also scaled up much better with the number of active CPU cores and showed much less degradation with the number of connections than Jool according to our measurements [4]. However, we have also experienced that the performance of iptables highly depends on certain parameters.

The aim of our current paper is to investigate how the performance of iptables depends on various settings, and also to examine what kind of tradeoffs exist, and thus recommend optimal settings depending on the actual performance needs and hardware parameters of the ISPs.

The remainder of this paper is organized as follows. In Section II, we make a survey how iptables is used in the current research papers and how its performance is analyzed and/or optimized. In Section III, we give a short summary of the state of the art methods for measuring the performance of stateful NAT44 gateways. In Section IV, we overview some relevant details of iptables including its tunable parameters and their recommended values as well as how they influence the memory consumption of iptables. In Section V, we disclose our measurements and their results. In Section VI, we discuss our results and give our recommendations to optimize the performance of iptables. Section VII is an additional case study in which we examine the performance of nftables. Section VIII concludes our paper.

## II. RELATED WORK

### A. Peer-reviewed Papers

We have surveyed, how iptables appears in research papers from the latest years. We found that it is usually mentioned as a firewall and not as a stateful NAT44 solution. And the methods used for measuring its performance does not comply with the relevant IETF RFCs, please see their requirements in Section III.A.

DOI: 10.36244/ICJ.2023.1.6

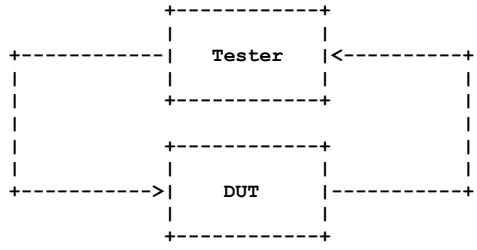



Fig. 1. Test setup for benchmarking network interconnect devices. [9]

For example, Melkov et al [3] compared the performance of iptables and nftables using very high number of rules (up to several times 10,000). In contrast with the common view, they have found that iptables significantly outperformed nftables. Depending on the actually examined chain, the throughput of nftables significantly deteriorated around 5,000 or 10,000 rules, whereas iptables could sustain a good performance up to 20,000 or 40,000 rules. We note that they measured "TCP throughput" using iperf, and displayed the results in Mbps.

Gandotra and Sharma [5] also measured the firewall performance of iptables using 200, 500, 1000, 5,000, and 10,000 number of rules, TCP traffic with 1024 bytes packets size, multiple packet rates starting from 1,000pps increased by 1,000pps steps to 8,000pps, and test durations of 30s and 120s. As for measurement tool, they used D-ITG (Distributed Internet Traffic Generator).

Taga at al. [6] used iptables for testing their firewall traversal method. As for measurement method, they downloaded HTML files with different sizes and measured the download time.

### B. Other Sources

In order to find more closely related sources to our topic, we have lowered the bar and did not require peer-reviewed papers.

Thus, we found a really closely related writing of Andree Toonk [7]. One of his tests was a stateful NAT44 performance measurement using a single iptables rule and 10,000 network flows. For the measurements, he used a DPDK-based packet generator, but the exact details of the measurements (how the bidirectional traffic was generated) are not disclosed. Using two 3.2GHz Intel Xeon Gold 5218 CPUs (in all 64 cores were available using hyper-threading) he managed to achieve a total of 5.9Mpps using bidirectional traffic. It does not turn out, if it was a lossless rate or not.

Whereas the above result is not bad, it definitely shows that there is room for performance optimization, as we achieved 5.3Mpps using only 16 cores of a 2.1GHz Intel Xeon E5-2683 v4 CPU even though we handled 1.56M connections (instead of only 10k) [4]. We note that our result is RFC 2544 [9] compliant throughput (non-drop rate). According to our measurements, the performance of iptables scaled up quite well with the number of CPU cores: when 4M connections were used and the number of active CPU cores was increased from 1 to 16, its *maximum connection establishment rate* (please refer to Section III.C) and *throughput* scaled up from 223.5kcps (connections per second) and 414.9kfps (frames per second) to 2,383kcps and 4,557kfps, respectively, thus the increase was more than tenfold [4]. We have also examined, how the performance of iptables degrades with the number of connections. In the range where we could increase the *hash table size* (please refer to Section IV.A) proportionally with the number of connections, the performance of iptables degraded only slightly with the 64-fold increase of the number of connection: when the number of connections were increased from 1.56M to 100M, its maximum connection establishment rate and throughput decreased from 2.406Mcps and 5.326Mfps to 2.237Mcps and 4.516Mfps, respectively. However, the degradation was more significant, when the built-in limitations of iptables prevented us from increasing the *hash table size* proportionally with the number of connections [4]. This is why we believe that it is worth examining how to optimize the parameters of iptables to provide ISPs with a high performance stateful NAT44 solution.

Theoretically, the reimplementation of iptables in eBPF could significantly outperform the native iptables. However, the measurement results of Massimo Tumolo show that it happens only if the number of the rules is above 100 [8]. It can happen, if iptables is used as a firewall. However, in our case, iptables is used as a stateful NAT44 gateway. Here the number of rules is very low (one or a few).

## III. BENCHMARKING METHODOLOGY FOR STATEFUL NAT44 GATEWAYS

### A. Benchmarking Methodology for Network Interconnect Devices

There is a long established benchmarking methodology for network interconnect devices defined by a series of IETF (Internet Engineering Task Force) RFCs. Commercial network performance tester vendors follow the requirements of RFC 2544 [9] for more than two decades. Its aim is to facilitate the measurement of the performance of network interconnect devices in an objective way. To that end it defines the most important conditions of the measurements to prevent gaming (or tricking or more openly: cheating), including:

- Test setup
- DUT (Device Under Test) settings (it may not be optimized for the given task)
- Test frame format and frame sizes (e.g. for Ethernet: 64, 128, 256, 512, 1024, 1280, 1518 bytes)
- Measurement procedures (throughput, latency, frame loss rate, back-to-back frames, system recovery, reset)
- Duration of the test (minimum 60s for throughput test)
- Requirement of testing with bidirectional traffic
- Usage of UDP as transport layer protocol
- Testing with a single IP address pair and also with 256 destination networks when routers are benchmarked.

As for test setup, the one shown in Fig. 1 should be used by default. Although the arrows are unidirectional, bidirectional traffic should be used as written above.

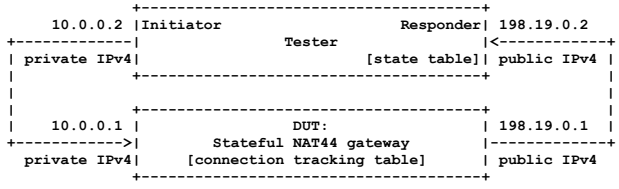



Fig. 2. Test setup for benchmarking stateful NAT44 gateways. [14]

From among the performance metrics, *throughput* is the most important one for us. It is defined as the highest constant frame rate, at which all frames can be forwarded by the DUT. Its measurement procedure requires that *test frames* are transmitted by the Tester through the DUT in both directions at least for 60 seconds at a constant frame rate, and the Tester counts the number of received test frames:

- If the number of the received test frames equals the number of the transmitted test frames then the frame rate is increased, and the test is rerun.
- If the number of the received test frames is less than the number of the transmitted test frames then the frame rate is decreased, and the test is rerun.

Whereas this wording facilitates various search algorithms, usually a binary search is executed using 0 and the maximum frame rate for the media as the starting interval.

Theoretically, it can be said that RFC 2544 is IP version independent, but in practice, it uses IPv4 addresses. The media types for which it defined the maximum frame rates in its appendix, also show its age.

As time passed by, the Benchmarking Working Group of IETF has produced further important RFCs. One of them is RFC 4814 [10]. It requires the usage of pseudorandom port numbers with uniform distribution over the following ranges:

- Source port number range: 1,024–65,535
- Destination port number range: 1–49,151

Without that requirement, the very same test frames could be sent, which was very convenient for the tester vendors, but it did not reflect the nature of the Internet traffic. Moreover, pseudorandom port numbers are necessary to support RSS (Receive-Side Scaling, also called multi-queue receiving) [11], because if the same source and destination IP addresses are used for each packet, then only the port numbers can ensure entropy for the hash function to distribute the interrupts of packet arrivals among the several cores of a contemporary CPU.

We note that there were two further important RFCs published. RFC 5180 [12] is mainly an IPv6 update regarding IPv6 specificities, but it also contains maximum frame rates for some media types being contemporary at the time of its writing. However, it excludes IPv6 transition technologies from its scope. They are covered by RFC 8219 [13]. It has kept the requirement of testing with bidirectional traffic, but it also introduces testing with single directional traffic as an optional measurement. We believe that the asymmetry of the amount of Internet traffic in download and upload directions is a good rational for testing also with unidirectional traffic, and it is worth extending it to the benchmarking of stateful NAT44 gateways, too.

### *B. Problems with Benchmarking Stateful NAT44 Gateways*

#### *1) Problems with the Feasibility of RFC Compliant Tests*

As for benchmarking stateful NAT44 gateways, we are faced with multiple problems. RFC 2544 requires testing with bidirectional traffic, whereas RFC 4814 requires the usage of pseudorandom port numbers with uniform distribution from the above mentioned ranges.

It can be easily calculated that the number of potential source port number destination port number combinations is more than three billion and it means so many network flows, thus potential entries in the connection tracking table of the stateful NAT44 gateways. Therefore, literally following this requirement in the private to public direction could exhaust the capacity of the connection tracking table of the DUT.

As for sending traffic in the public to private direction using pseudorandom port numbers, it would result in sending a lot of frames that do not belong to any existing connection, thus the stateful NAT44 gateway would simply discard them and the throughput test would fail.

#### *2) Problems with the Widely-used Measurements*

Researchers were creative enough to accommodate to the limitation of NAT44 that connections may be initiated only from the private side. They put the iperf or D-ITG server on the public side and thus the measurement was feasible. However, this type of measurement has serious limitations. To examine them, let us see, what happens (and what may happen) during the execution of a test. At the beginning of the test, most of the test frames sent form the private side result in new connections in the stateful NAT44 gateway. As time elapses, its connection tracking table has more and more connections and thus the proportion of the test frames that belong to an existing connection will increase. The proportion of the test frames resulting in new connections will likely decrease significantly and it may even become zero. The progress of this change depends on several factors including:

1. how the client is programmed (how many different network flows are used and what policy it follows to send a test frame that belongs to an already used or a new flow)
2. the connection timeout time of the stateful NAT44 gateway
3. the size and policy of the connection tracking table of the stateful NAT44 gateway.

The main problem with this type of measurement is that the test traffic is a kind of a mix, in which the proportion of the ingredients (frames resulting in a new connection or not) varies with the time. It results in several negative consequences, including:

1. It is rather hard to tell, exactly what was measured: e.g., the connection setup performance or the frame forwarding performance of the DUT?
2. The results of measurements performed with different tools are likely not comparable.

3. It is hard to tell, what conditions are needed to achieve reproducible measurements.
4. It is not possible to measure some clear and well defined characteristics like bidirectional, download-only, and upload-only throughput or connection setup performance.

Our methodology offers remedy for all these problems.

### C. Our Methodology

We have defined a general methodology [14] suitable for the benchmarking of any stateful NATxy gateways using RFC 4814 pseudorandom port numbers, where x and y are in {4, 6}.

Now we give a brief introduction to the methodology using the example of the stateful NAT44.

The test setup is shown in Fig. 2. The DUT is the stateful NAT44 gateway, which has a connection tracking table. Its content, size, and replacement policy is unknown for the Tester. The Tester can influence or examine its content in indirect ways:

- The Tester can add a new connection to the connection tracking table by sending a test frame in the private to public direction with a new source port number destination port number combination.
- The Tester can check, if a given connection is present in the connection tracking table by sending a test frame belonging to the given connection in the public to private direction and verifying if the test frame arrives back.

There are two operations that can be performed by some out of band methods:

1. The timeout time of the connections can be set to any permitted value.
2. The entire content of the connection tracking table can be deleted.

Please refer to Section V.A, how these operations can be performed with iptables.

As the operation of the stateful NAT44 gateway is asymmetric, the operation of the Tester is also asymmetric.

The Initiator can send a test frame using any desired source port number destination port number combinations, but it uses restricted ranges to avoid the exhaustion of the capacity of the connection tracking table of the DUT. The size of the source port number range is larger (e.g. a few times 10,000) and the size of the destination port numbers is smaller (e.g. in the order of 10, 100, or 1000), and it can be used as a parameter to perform the measurements with different number of network flows. Please refer to our Internet Draft [14] for the rationale of the asymmetry of the sizes of the port number ranges. (The source and destination IP addresses have constant values and the protocol is always UDP.)

The Responder may not invent any flow identifiers, but it extracts the *four tuples* (source IP address, source port number, destination IP address, destination port number) from the received test frames and stores them in its *state table*. When it sends a test frame, it takes a four tuple from its state table (swaps source and destination), and thus it creates a valid test frame that belongs to an existing connection in the connection tracking table of the DUT.

To make testing possible, we have introduced the *preliminary test phase*. During this phase only the Initiator sends test frames. The DUT registers the new connections into its connection tracking table, translates the test frames and forwards them to the Responder. Thus, the connection tracking table of the DUT and the state table of the Responder are initialized, and in the *real test phase*, the Responder is able to send valid test frames.

To achieve clear and repeatable measurements, we use two extreme situations that we can simply ensure:

1. All test frames create a new connection during the preliminary test phase.
2. Test frames never create a new connection in the real test phase.

We achieve them by using

- large enough and empty connection tracking table for each test
- pseudorandom enumeration of all possible source port number destination port number combinations in the preliminary test phase
- a properly high timeout value in the DUT (higher than the time duration from the beginning of the preliminary test phase to the end of the real test phase including timeout).

To quantify the connection setup performance of the DUT, we have introduced the *maximum connection establishment rate* as a new metric. It is the highest constant frame rate at which the DUT is able to process all test frames in the preliminary test phase. (Each test frame is successfully translated and a new connection is created in the connection tracking table.) Its measurement procedure is very simple to that of the throughput, the details can be found in our Internet Draft.

All "classic" measurements (throughput, latency, frame loss rate, etc.) can be performed in the real test phase. To that end, first, the preliminary test phase has to be executed using a frame rate safely lower than the measured connection establishment rate. Then comes the real test phase with the desired measurement.

The other side of connection establishment is connection tear down. We defined *connection tear down rate* to quantify the connection tear down performance of the DUT. Is short, it is measured as follows. First, $N$ number of connections are loaded into the connection tracking table of the DUT. Then the entire content of the connection tracking table is deleted, and its $T$ deletion time is measured. The connection tear down rate is calculated as: $N/T$. It is measured for different values of $N$.

We give more details in Section V.B.4, were we describe our test to measure the connection tear down rate of iptables.

## IV. IMPORTANT DETAILS OF IPTABLES

First of all, iptables does *connection tracking* not only for stateful protocols (TCP), but also for stateless ones (UDP, ICMP). The connection tracking system of iptables is

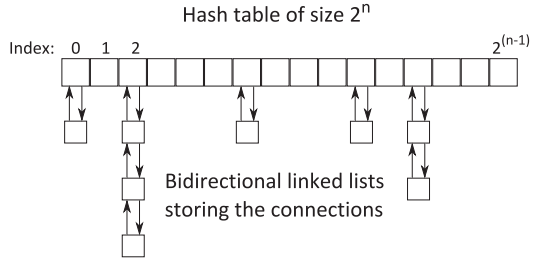



Fig. 3. Implementation of the connection tracking table of iptables using a hash table and bidirectional linked lists (based on [2]).

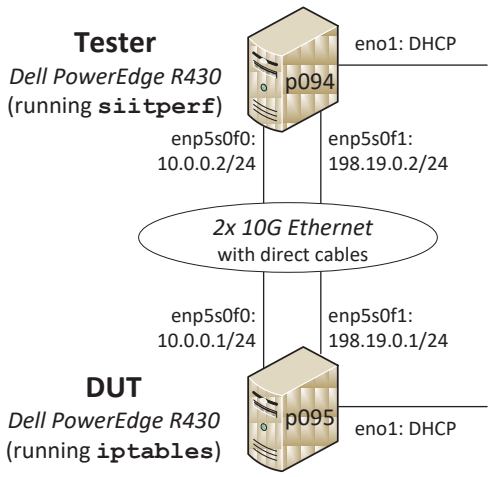



Fig. 4. Topology of the stateful NAT44 test system.

implemented using *hashing* to ensure efficient lookups. As for UDP, the source and destination IP addresses, as well as the source and destination port numbers are parts of the hash tuple. The size of the hash table is a power of 2. Hash collisions are handled that the *connection tracking entries* are stored as the elements of bidirectional linked lists starting from the *hash table* entries, as shown in Fig. 3.

### A. Parameters to Tune

The *size of the hash table* fundamentally influences the efficiency of hashing and thus also the performance of iptables. Its default size is automatically determined on the basis of the memory size of the Linux system. It can be read or modified by reading or writing the **`/sys/module/nf_conntrack/parameters/hashsize`** file.

The *maximum number of the connection tracking entries* is another important parameter. It can be read or written using **`sysctl`** as **`net.netfilter.nf_conntrack_max`**.

Traditionally, the values of the above two parameters are set as: **`hashsize=nf_conntrack_max/8`** [15], [16]. It means that the *average* length of the linked list starting from the hash table entries may be up to 8. However, this is just a convention, and in our current paper, we examine what value is worth using.

We note that newer systems use 4 instead of 8 when they set the default values, please refer to Table VI.

As for timeout, iptables handles several different timeouts for the various states of TCP, two different ones for UDP and one for ICMP. Now for us it is enough to handle (using **`sysctl`**): **`net.netfilter.nf_conntrack_udp_timeout`**.

### B. Memory Consumption

The memory consumption of the *hash table* is likely 8 bytes per entry, as 64 bits are required to store a pointer pointing to the first element of the linked list, and we also confirm it by measurements in Section V.C.1.

As for the memory consumption of the *connection tracking entries*, we have found several different values.

1. Section 3.7 of [17] (a document from 2007) states that "each tracked connection eats about 350 bytes of non-swappable kernel memory".
2. Another source from 2009 stated that as each object was 304 bytes in size and 13 of them fitted in a 4096 bytes page, thus 144 bytes was wasted and so the effective memory consumption was about 316 bytes per **`nf_conntrack`** object [18].
3. We found in a commit message from 2016 that "increases struct size by 32 bytes (288 -> 320), but it is the right thing, else any attempt to (re-)arrange nf_conn members by cacheline won't work." [19]

Therefore, we have determined the effective memory consumption of the *connection tracking entries* by measurement in Section V.C.2.

## V. Our Benchmarking Measurements

### A. Measurement System

The topology of the measurement system is shown in Fig. 4. Both the Tester and the DUT were Dell PowerEdge 430 servers. Each one had two 2.1GHz Intel Xeon E5-2683 v4 CPUs, 384GB 2400MHz DDR4 RAM and an Intel 10GbE dual port X540 network adapter. They were interconnected by direct cables to avoid frame loss. Debian 9.13 Linux operating system with 4.9.0-16-amd64 kernel was used on both servers. The CPU clock frequencies of both servers were set to fixed 2.1GHz using the **`tlp`** Debian package and hyper-threading was switched off to ensure reliable results. Only CPU cores 0-15 of the DUT were online. The version of iptables was 1.6.0. The test system was the same as we used for our measurements in Section 2 of [4], thus our results are directly comparable.

For performing measurements, we used the **`siitperf`** [20] RFC 8219 compliant SIIT and stateful NAT64 / NAT44 tester program documented in [21].

We note that **`siitperf`** was developed for research purposes and it is a collection of *binary programs* (written in C/C++ using DPDK) that perform elementary tests, and **`bash`** *shell scripts* that have to be tailored to the measurement environment. As for stateful tests, they use **`ssh`** to execute some scripts on the DUT e.g., to empty the connection tracking table of the DUT. For our current tests, we used two such scripts on the **`p095`** server. The **`set-iptables-hcpx`** script was used to set the iptables rule, the size of the hash table,

TABLE I
MAXIMUM CONNECTION ESTABLISHMENT RATE OF IPTABLES AS A FUNCTION OF THE HASH TABLE SIZE, 40M CONNECTIONS, 16 CPU CORES

| Hash table size | 2^26 | 2^25 | 2^24 | 2^23 | 2^22 | 2^21 |
|---|---|---|---|---|---|---|
| Number of connections / hash table size | 0.5960 | 1.1921 | 2.3842 | 4.7684 | 9.5367 | 19.0735 |
| Error of binary search (cps) | 100 | 100 | 100 | 100 | 100 | 100 |
| Median (cps) | 2,263,732 | 2,075,195 | 1,696,411 | 1,231,993 | 780,090 | 440,124 |
| Minimum (cps) | 2,203,063 | 2,044,493 | 1,624,938 | 1,216,734 | 764,098 | 421,813 |
| Maximum (cps) | 2,359,680 | 2,125,549 | 1,750,061 | 1,251,037 | 797,912 | 445,617 |
| Median / previous median | - | 0.92 | 0.82 | 0.73 | 0.63 | 0.56 |

TABLE II
THROUGHPUT OF IPTABLES AS A FUNCTION OF THE HASH TABLE SIZE, 40M CONNECTIONS, 16 CPU CORES, BIDIRECTIONAL TRAFFIC

| Hash table size | 2^26 | 2^25 | 2^24 | 2^23 | 2^22 | 2^21 |
|---|---|---|---|---|---|---|
| Number of connections / hash table size | 0.5960 | 1.1921 | 2.3842 | 4.7684 | 9.5367 | 19.0735 |
| Error of binary search (fps) | 200 | 200 | 200 | 200 | 200 | 200 |
| Median (fps) | 4,252,197 | 4,068,973 | 3,683,713 | 3,150,632 | 2,445,677 | 1,719,848 |
| Minimum (fps) | 4,174,436 | 3,899,872 | 3,624,876 | 2,999,876 | 2,428,586 | 1,716,672 |
| Maximum (fps) | 4,282,348 | 4,103,502 | 3,719,360 | 3,187,622 | 2,459,836 | 1,728,148 |
| Median / previous median | - | 0.96 | 0.91 | 0.86 | 0.78 | 0.70 |

TABLE III
CONNECTION TEAR DOWN RATE OF IPTABLES AS A FUNCTION OF THE HASH TABLE SIZE, 40M CONNECTIONS, 16 CPU CORES

| Hash table size | 2^26 | 2^25 | 2^24 | 2^23 | 2^22 | 2^21 |
|---|---|---|---|---|---|---|
| Number of connections / hash table size | 0.5960 | 1.1921 | 2.3842 | 4.7684 | 9.5367 | 19.0735 |
| Filled table deletion time – Median (s) | 158.51 | 154.07 | 153.52 | 150.61 | 150.12 | 149.76 |
| Filled table deletion time – Minimum (s) | 157.63 | 152.81 | 152.21 | 149.94 | 148.62 | 148.43 |
| Filled table deletion time – Maximum (s) | 158.89 | 154.72 | 153.69 | 151.50 | 150.41 | 150.13 |
| Empty table deletion time – Median (s) | 8.00 | 4.15 | 2.24 | 1.26 | 0.78 | 0.55 |
| Empty table deletion time – Minimum (s) | 7.99 | 4.14 | 2.22 | 1.25 | 0.77 | 0.54 |
| Empty table deletion time – Maximum (s) | 8.02 | 4.17 | 2.25 | 1.28 | 0.81 | 0.57 |
| Net connection deletion time (s) | 150.51 | 149.92 | 151.29 | 149.35 | 149.34 | 149.21 |
| Connection tear down rate (cps) | 265,763 | 266,809 | 264,402 | 267,827 | 267,845 | 268,079 |

the maximum number of connections, and the UDP timeout value before each elementary test. The **`del-iptables`** script deleted the iptables rule and the content of the connection tracking table (by the removal of the kernel modules) after each elementary test. They are available on GitHub [22].

### B. Performance Measurements

#### 1) Aim, Parameters, and Types of Tests

We aimed to examine, how the ratio of the number of connections and the hash table size influences the performance of iptables.

Gapon [16] recommended 4,194,304 as the upper limit for number of connections for a highly loaded NAT server and 524,288 for hash table size. We decided to use rather 40M connections, because we wanted to test iptables under really demanding condition. We achieved this number of port number combinations by using 40,000 source port numbers and 1,000 destination port numbers.

To be able to handle 40M connections, the first appropriate power of 2 for the maximum number of connections is $2^{26}$=67,108,864. As for hash table size, first, we used the same value and then we halved it five times, thus the final tested value was $2^{21}$=2,097,152.

We set the UDP timeout to 10,000 seconds, to guarantee its high enough value for all tests.

We measured the maximum connection establishment rate, the throughput, and connection tear down rate with each hash table size. All measurements were performed 10 times to get reliable results.

#### 2) Maximum Connection Establishment Rate

The maximum connection establishment rate of iptables as a function of the hash table size is shown in Table I. (The "error of binary search" value expresses the stopping criterion for the binary search. It stops, when:

higher_limit – lower_limit <= error.)

Although the independent variable is the hash table size (shown as a power of 2), what really helps to understand the behavior of the system is the average number of connections hashed to the same hash table entry, that is *the average length of the linked lists*. It is computed as the number of connections per hash table size. In the first step, it increases from about 0.6 to about 1.2, and the median of the maximum connection establishment rate decreases only 8%. However, its further doubling causes more and more radical decrease of the median.

#### 3) Throughput

The throughput of iptables as a function of the hash table size is shown in Table II. It behaves similarly to the maximum connection establishment rate in the sense that the doubling of the average length of the linked list causes more and more radial decrease of the performance when it becomes significantly higher than 1, but the measure of the deterioration is lower.

#### 4) Connection Tear Down Rate

Having no better way to measure the connection tear down rate, we used an aggregate measurement that $N$ number of connections were loaded into the connection tracking table of iptables and then the entire table was deleted and the $T$ duration of the deletion was measured. However, then the deletion time

TABLE IV
MEMORY CONSUMPTION OF THE HASH TABLE OF IPTABLES AS A FUNCTION OF THE HASH TABLE SIZE

| Hash table size | 2^26 | 2^25 | 2^24 | 2^23 | 2^22 | 2^21 |
|---|---|---|---|---|---|---|
| Median (kB) | 523,560 | 261,514 | 130,238 | 64,964 | 32,444 | 16,248 |
| Minimum (kB) | 521,816 | 260,208 | 129,024 | 63,436 | 30,824 | 15,368 |
| Maximum (kB) | 527,092 | 263,284 | 132,108 | 66,108 | 33,132 | 16,896 |
| Bytes per hash table entry | 7.99 | 7.98 | 7.95 | 7.93 | 7.92 | 7.93 |

TABLE V
MEMORY CONSUMPTION OF IPTABLES WITH 40M CONNECTION TRACKING ENTRIES AS A FUNCTION OF THE HASH TABLE SIZE

| Hash table size | 2^26 | 2^25 | 2^24 | 2^23 | 2^22 | 2^21 |
|---|---|---|---|---|---|---|
| Number of connections / hash table size | 0.5960 | 1.1921 | 2.3842 | 4.7684 | 9.5367 | 19.0735 |
| Median (kB) | 15,056,120 | 15,056,038 | 15,056,018 | 15,055,478 | 15,053,256 | 15,054,764 |
| Minimum (kB) | 15,055,296 | 15,050,388 | 15,053,528 | 15,050,028 | 15,050,876 | 15,051,868 |
| Maximum (kB) | 15,058,672 | 15,057,412 | 15,060,400 | 15,057,376 | 15,057,096 | 15,057,196 |
| Bytes per hash table entry | 385.44 | 385.43 | 385.43 | 385.42 | 385.36 | 385.40 |

contained the time necessary to delete an empty connection tracking table, as well as the command execution and communications latencies, too. To make our results more accurate, we have also measured the duration of the deletion of an empty table, which also contained the command execution and communications latencies. Thus their difference contains only the time spent by the deletion of the $N$ number of connections, which is called as *net connection deletion time* in Table III. This value is nearly the same independently form the hash table size. Thus, the connection tear down rate is also independent from the size of the hash table.

### C. Memory Consumption Measurements

#### 1) Hash Table

To measure the memory consumption of the hash table, we set various hash table sizes, and checked, how the memory usage of the Linux systems changed. We considered the "used" value in the output of the **`free`** Linux command. We could not set arbitrarily small hash table size: if we tried setting it to a smaller value than 512 entries, then it was set to 512.

As with the other measurements, we set each size 10 times (including 512) and recorded the amount of the used memory of the Linux system with a script. Then we subtracted the memory usage measured with 512 entries from all the other memory usage values. We calculated median, minimum and maximum of the results, and finally, we computed the memory consumption per hash table entry using the median values for the calculation. The results are shown in Table IV. Of course, the 512*8=4,096 bytes memory consumption of the hash table causes some small error, but the results can still confirm that the *memory consumption of the hash table is 8 bytes per entry*.

#### 2) Connection Tracking Entries

To measure the memory consumption of the connection tracking entries, we inserted 40M connections into the connection tracking table using safely lower frame rates than the maximum connection establishment rate for the given hash table size. Then we recorded the memory usage of the Linux system, next, deleted the content of the connection tracking table, and finally, recorded the memory usage of the Linux system again.

As with the other measurements, we performed the tests with each connection tracking table size 10 times.

We calculated the difference of the memory usage of the Linux system when the connection tracking table had 40M entries and when it was empty. This difference is the memory consumption of the 40M connection tracking entries. The results are shown in Table V. The *memory consumption of the 40M connection tracking entries is independent from the size of the hash table*, and on average, a *single connection tracking entry consumes 385.4 bytes*. The results are very stable: the difference of the maximum and minimum is always less than 0.1% of the median.

## VI. DISCUSSION OF THE RESULTS AND OUR RECOMMENDATION FOR SETTING HASH TABLE SIZE

Our performance measurements showed that value of the *number of connections per hash table size* is a very important parameter that highly influences the performance of iptables. This parameter gives the average length of the linked lists starting from the entries of the hash table. Both the maximum connection establishment and the throughput of iptables seriously deteriorates when this number becomes significantly higher than 1. But the connection tear down rate does not depend on it at all.

We have also checked the "price" of the performance and we found that the memory consumption of the hash table is proportional to its size: each entry requires 8 bytes. However, the memory consumption of the connection tracking entries does not depend on the size of the hash table, and each entry occupies approximately 385.4 bytes. The orders of magnitude of these two numbers suggest us that the memory consumption of the hash table entries is practically negligible compared to the memory consumption of the connection tracking entries. With other words: the 40M connection tracking entries occupy about 15GB RAM independently from the hash table size as shown in Table V, whereas the memory consumption of the hash table itself varies between 0.5GB and 16MB as shown in Table IV, thus the latter is practically negligible.

Therefore, we definitely recommend to abandon using the **`hashsize=nf_conntrack_max/8`** convention and rather use **`hashsize=nf_conntrack_max`** to increase the performance of iptables significantly. Of course, now arises the question of using **`hashsize=nf_conntrack_max*n`**, where $n > 1$. We

TABLE VI
DEFAULT AND MAXIMUM ALLOWED VALUES FOR **HASHSIZE** AND **NF_CONNTRACK_MAX** AS A FUNCTION OF MEMORY SIZE, IPTABLES 1.6.0

| Computer memory size (GB) | 1 | 2 | 8 | 384 |
|---|---|---|---|---|
| Default **hashsize** | 7,680 | 16,384=2^14 | 65,536=2^16 | 65,536=2^16 |
| Default **nf_conntrack_max** | 30,720 | 65,536=2^16 | 262,144=2^18 | 262,144=2^18 |
| Maximum possible **hashsize** | 33,554,432=2^25 | 67,108,864=2^26 | 268,435,456=2^28 | 268,435,456=2^28 |
| Maximum possible **nf_conntrack_max** | 1,073,741,824=2^30 | 1,073,741,824=2^30 | 1,073,741,824=2^30 | 1,073,741,824=2^30 |

TABLE VII
PERFORMANCE OF NFTABLES 0.9.0-2, 2^26 HASH TABLE SIZE, 40M CONNECTIONS, 16 CPU CORES

| Performance metric | Max. conn. est. rate (cps) | Throughput (fps) |
|---|---|---|
| Error of bin. search | 100 | 200 |
| Median | 228,222 | 835,544 |
| Minimum | 225,683 | 824,804 |
| Maximum | 229,198 | 840,428 |

did not do tests with n=2, 4, 8, etc. but our results show that the expectable gain is much less. When using iptables as a stateful NAT44 gateway to forward Internet traffic, there are a high number of packets transferred per session. Thus, throughput is the dominant one from among our three used performance metrics. Examining Table II, we can see that there is only 4% performance difference between the first two columns. And considering the observable trends, it is likely to be even less, if the size if the hash table is further increased. And the increase of the hash table size has a built in limit. Theoretically, the default and the allowed maximum values for **hashsize** and **nf_conntrack_max** depend on the RAM size of the computer [15]. Our measurements show that it is true for very small RAM sizes (e.g. 1GB or 2GB), but the values do not change from 8GB and to 384GB RAM size. We used the **mem=*n*GB** kernel command line parameter to limit the available memory for testing. Our results in Table VI show that the default and maximum values are the same for 8GB and 384GB.

## VII. TESTING THE PERFORMANCE OF NFTABLES

We have also tested the maximum connection establishment rate and throughput performance of nftables. We used the same test system as shown in Fig. 4, but Debian 10.13 with 4.19.0-20-amd64 kernel was used on the DUT. The version of nftables was: 0.9.0-2. We tested its performance only at the "optimal" working point, that is, using our recommended setting: **hashsize=nf_conntrack_max**.

The results are shown in Table VII. Comparing the results with that of iptables (shown in Table I and Table II), we can see that the median maximum connection establishment rate of nftables (228,222cps) is about one tenth of the median maximum connection establishment rate of iptables (2,263,732cps), whereas the median throughput of nftables (835,544fps) is about one fifth of the median throughput of iptables (4,252,197fps) measured under the same conditions. Therefore, we conclude that nftables may not replace iptables in the application scenarios where a stateful NAT44 gateway of a CGN system has to handle a high number of connections with high performance.

## VIII. CONCLUSION

We have measured the maximum connection establishment rate, throughput and connection tear down rate as well as the memory consumption of iptables as a function of the hash table size using always 40 million connections to determine the optimal value for the ratio of the number of connections and the hash table size and/or any possible tradeoff.

We conclude that the long established convention of **hashsize=nf_conntrack_max/8** should be replaced by the **hashsize=nf_conntrack_max** rule to increase the performance of iptables to a high extent.

We have also shown that nftables may not replace iptables in the application scenarios where a stateful NAT44 gateway of a CGN system has to handle a high number of connections with high performance, because iptables achieved about ten times higher maximum connection establishment rate and about five times higher throughput than nftables.

## ACKNOWLEDGMENT

The experiments were carried out by remotely using the resources of NICT StarBED, 2–12 Asahidai, Nomi-City, Ishikawa 923–1211, Japan.

The authors would like to thank Shuuhei Takimoto for the possibility to use StarBED, as well as to Tsukasa Nishita Makoto Yoshida for their help and advice in StarBED usage related issues.

Gábor Lencse thanks the National Institute of Information and Communications (NICT), Japan for their support of his stay at the Research Laboratory of Internet Initiative Japan, where his research topic was the performance analysis of stateful NAT64 / NAT44 implementations.

## REFERENCES

[1] M. Nikkhah, R. Guérin, "Migrating the Internet to IPv6: An exploration of the when and why", *IEEE/ACM Trans. Netw*., vol. 24, no. 4, pp. 2291–2304, Apr. 2016, DOI: 10.1109/TNET.2015.2453338

[2] P. N. Ayuso, "Netfilter's connection tracking system", *Login: The Usenix Magazine*, vol. 31, no. 3, (2006) pp. 34–39. [Online]. Available: https://www.usenix.org/system/files/login/articles/892-neira.pdf

[3] D. Melkov, A. Šaltis and Š. Paulikas, "Performance Testing of Linux Firewalls", *2020 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream)*, 2020, pp. 1–4, DOI: 10.1109/eStream50540.2020.9108868.

[4] G. Lencse, "Scalability of IPv6 transition technologies for IPv4aaS", Internet Draft, Oct 23, 2022, draft-lencse-v6ops-transition-scalability-04 [online], available: https://datatracker.ietf.org/doc/html/draft-lencse-v6ops-transition-scalability-04

[5] N.Gandotra, L.S. Sharma, "Performance evaluation and modelling of the Linux firewall under stress test", In: Singh, P., Kar, A., Singh, Y., Kolekar, M., Tanwar, S. (eds) Proceedings of ICRIC 2019, *Lecture Notes in Electrical Engineering*, vol 597. Springer, DOI: 10.1007/978-3-030-29407-6_54

[6] K. Taga, J. Zheng, K. Mouri, S. Saito, E. Takimoto, "Firewall traversal method by pseudo-TCP encapsulation", *IEICE Transactions on Information and Systems*, 2022, vol. E105.D, no. 1, pp. 105–115, 2022, DOI: 10.1587/transinf.2021EDP7050

[7] A. Toonk, "Linux kernel and measuring network throughput", personal blog, [Online], available: https://toonk.io/linux-kernel-and-measuring-network-throughput/index.html

[8] M. Tumolo, "Towards a faster Iptables in eBPF", MSc thesis, Politechnico di Torino, 2017-2018, [online], available: https://webthesis.biblio.polito.it/secure/8475/1/tesi.pdf

[9] S. Bradner, and J. McQuaid, "Benchmarking methodology for network interconnect devices", *IETF RFC 2544*, 1999. DOI: 10.17487/RFC2544.

[10] D. Newman, T. Player, "Hash and stuffing: Overlooked factors in network device benchmarking", *IETF RFC 4814*, 2008. DOI: 10.17487/RFC4814

[11] T. Herbert, W. de Bruijn, "Scaling in the Linux networking stack", [Online]. Available: https://www.kernel.org/doc/Documentation/networking/scaling.txt

[12] C. Popoviciu, A. Hamza, G. V. de Velde, and D. Dugatkin, "IPv6 benchmarking methodology for network interconnect devices", *IETF RFC 5180*, 2008, DOI: 10.17487/RFC5180.

[13] M. Georgescu, L. Pislaru, and G. Lencse, "Benchmarking methodology for IPv6 transition technologies", *IETF RFC 8219*, Aug. 2017, DOI: 10.17487/RFC8219

[14] G. Lencse, K. Shima, "Benchmarking methodology for stateful NATxy gateways using RFC 4814 pseudorandom port numbers", Internet Draft, Sep 24, 2022, draft-ietf-bmwg-benchmarking-stateful-00 [Online], available: https://datatracker.ietf.org/doc/html/draft-ietf-bmwg-benchmarking-stateful-00

[15] H. Eychenne, "Conntrack tuning: Netfilter conntrack performance tweaking, v0.8", 2008, [Online], available: https://wiki.khnet.info/index.php/Conntrack_tuning

[16] V. Gapon, "Tuning nf_conntrack", personal blog, [Online], available: https://ixnfo.com/en/tuning-nf_conntrack.html

[17] H. Welte, "Netfilter/iptables FAQ" 2007, [Online], available: https://www.netfilter.org/documentation/FAQ/netfilter-faq.html

[18] J. Leach, "Netfilter conntrack memory usage", [Online], available: https://johnleach.co.uk/posts/2009/06/17/netfilter-conntrack-memory-usage/

[19] P. N. Ayuso, "[06/26] netfilter: conntrack: align nf_conn on cacheline boundary", commit message, 2016, [Online], available: https://patchwork.ozlabs.org/project/netdev/patch/1467815048-2240-7-git-send-email-pablo@netfilter.org/

[20] G. Lencse, "Siitperf: an RFC 8219 compliant SIIT and stateful NAT64/NAT44 tester", free software under GPLv3 license, source code, [Online], available: https://github.com/lencsegabor/siitperf

[21] G. Lencse, "Design and implementation of a software tester for benchmarking stateful NATxy gateways: theory and practice of extending siitperf for stateful tests", *Computer Communications*, vol. 172, no. 1, pp. 75-88, Aug. 1, 2022, DOI: 10.1016/j.comcom.2022.05.028

[22] G. Lencse, "DUT settings for benchmarking iptables and nftables" [Online], available: https://github.com/lencsegabor/DUT-settings-iptables-nftables

**Gábor Lencse** received his M.Sc. and Ph.D. degrees in computer science from the Budapest University of Technology and Economics, Budapest, Hungary in 1994 and 2001, respectively.
He works for the Department of Telecommunications, Széchenyi István University, Győr, Hungary since 1997. Now, he is a Professor. He is also a part time Senior Research Fellow at the Department of Networked Systems and Services, Budapest University of Technology and Economics since 2005. His research interests include the performance and security analysis of IPv6 transition technologies. He is a co-author of RFC 8219 and RFC 9313.

**Keiichi Shima** is a deputy director at the Research Institute of Advanced Technology of SoftBank Corp. His research field is the Internet and mobile network, including designing and implementing communication protocols, operation technologies, network security, and so forth. He also works as a board member of the WIDE project operating a nation wide research network in Japan.

# Dynamic Distributed Monitoring for 6LoWPAN-based IoT Networks

Basma Mostafa*†, Miklos Molnar†, Mohamed Saleh*, Abderrahim Benslimane‡, and Sally Kassem*§

***Abstract*—Mission-critical Internet of Things (IoT)-based networks are increasingly employed in daily and industrial infrastructures. The resilience of such networks is crucial. Given IoT networks' constantly changing nature, it is necessary to provide dependability and sustainability. A robust network monitoring can reinforce reliability, such that the monitoring mechanism adapts itself to real-time network instabilities. This work proposes *a proactive, dynamic, and distributed* network monitoring mechanism with monitor placement and scheduling for 6LoWPAN-based IoT networks intended for mission-critical applications. The proposed mechanism aims to ensure real-time monitoring coverage while respecting the limited and changing power resources of devices to prolong the network lifetime.**

***Index Terms*—IoT networks; Reliability; Monitor Scheduling; Dynamic; Proactive Monitoring; Critical Missions**

## I. INTRODUCTION

The Internet of Things (IoT) is a global network and service infrastructure composed of heterogeneous things with identities and physical and virtual attributes and seamlessly integrated into the Internet [29]. The IoT aims to offer various services by enabling things to be connected anytime, anyplace, with anything and anyone, and ideally using any network infrastructure. By connecting billions of things to the Internet, IoT created a plethora of applications that touch every aspect of human life, to name but a few: wearables, smart homes, smart cities, smart grids, and connected cars. IoT is present mainly in manufacturing, production, system monitoring, automation, and also in the Industrial IoT (IIoT) (often referred to as Industry 4.0) [7].

An essential category of IoT networks in industrial applications is the Low-Power and Lossy Network (LLN). Following the indications in RFC 7102 [43]: an LLN is a network of embedded devices with limited power, memory, and processing resources. LLNs are typically optimized for energy efficiency. They may use IEEE 802.15.4, which can be applied in IIoT, building automation, connected homes, healthcare, environmental monitoring, urban sensor networks, asset tracking, and more. However, the use of LLN in critical systems is challenging.

IoT networks have self-configuring capabilities and should be based on standard and interoperable protocols to foster smart, sustainable, and inclusive IoT services and products [11]. Unfortunately, they are often characterized by several challenges. The tight energy, memory, and processing constraints of the things and unreliable radio communication are naturally added to the difficulties of node failures, long-term network instability, security, and resource-exhaustion attacks [19, 32].

For resource-constrained IoT entities, minimizing the energy consumed for communication and computing is a primary constraint [42]. Moreover, there is an aggravating need to devise solutions that optimize energy and enhance IoT sustainability, which recently became a hot research area [30, 11]. Hence, the motivation for investigating the development of detailed protocol (re)design and usage to reduce energy consumption during normal operation and under Denial of Service (DoS) attacks, especially for domains where network robustness and safety requirements are crucial [19, 18].

The following section, Section II, describes the motivations behind our research in response to the requirements of mission-critical IoT solutions and those of a consequent monitoring mechanism. Section III reviews the state of the art and mentions the gap that our research fills. Section IV describes the essential elements of the original CGS scheduling algorithm for area coverage, and then it presents our adaptation for realizing IoT network monitoring. The experimentation of the proposed mechanism is shown in Section V. Finally, Section VI summarizes the conclusions of our and the possible future research directions.

## II. CHALLENGES & REQUIREMENTS

Several challenges are faced when deploying IoT solutions, (summarized in Table I).

Our research focuses on applications where the results are essential, and the mission has to be successful at any cost. Such applications are well-known as *mission-critical* since they deal with serious situations with high priorities for increased *reliability* and *network coverage*.

Examples of mission-critical applications are safety-oriented ones such as surveillance for safety and security applications [13]. IoT is ideal here since it can be successfully integrated within mission-critical systems deployed at locations where human presence is impossible due to human life's dangers. In such cases, gathering information can be done through IoT sensors and sent directly to the processing hubs [35] to detect failures and assess dangerous events. Thus, corrective, preventive, and rescue actions can be taken promptly.

Corresponding author: B. Mostafa (E-mail: basma.mostafa@lirmm.fr)
* Faculty of Computers & Artificial Intelligence, Cairo University, Cairo, Egypt
† LIRMM, Université de Montpellier, Montpellier, France
‡ LIA, Université d'Avignon, Avignon, France
§ Smart Engineering Systems Center, Nile University, Cairo, Egypt

DOI: 10.36244/ICJ.2023.1.7

TABLE I
IOT SOLUTIONS' DEPLOYMENT CHALLENGES

| |
|---|
| **Instability.** Unreliable, lossy channels with unpredictable bandwidth between things [33], and eventual node unreachability [37]. |
| **Limited network lifetime.** Lifetime should be maximized by incorporating duty-cycling mechanisms (*i.e.*, Active/sleep alternation by the turn on/off of the nodes' activity)[17]. |
| **Resource constraints.** Things have stringent resource constraints for energy, processing power, and memory of devices [33]. |
| **Mobility.** Mobile devices and highly dynamic network topology [14]. |
| **Vast number of heterogeneous devices.** The increasing number of connected devices produces scalability issues in data communication, networking, service provisioning, and management. |
| **vulnerability to security risks.** The shared wireless medium and access to the Internet alleviate the effect of security risks [21]. |
| **Denial of Service (DoS).** Also known as resource-exhaustion attack is one of the significant threats to availability, depriving users of services by consuming IoT nodes [41]. |

Other examples of mission-critical applications are:

- military applications such as intrusion in remote or hostile environments,
- environmental monitoring such as detecting the presence of methane and carbon monoxide gases in mines and triggering rescue protocols,
- disaster management, for instance, detecting radioactive and toxic gases in hostile environments,
- rescue operations, for instance, detecting fires, jostling in large smart stores, and triggering evacuation protocols,
- health monitoring to monitor chronic disease patients [8] and heart, panic, and epileptic-related attacks of drivers through the Internet of Vehicles (IoV) to prevent accidents [15].

For critical applications, the recovery time in case of network failure could be intolerable. Application robustness, fault avoidance, and recoverability of communicant objects in the network in uncertain information require effective defense mechanisms; weaknesses must be controlled and corrected *before* disrupting service provision. Thus, to prevent the deterioration of IoT systems and maintain a fault-tolerant solution, effort should be invested in developing *proactive*, efficient monitoring of the network, and fast correction mechanisms [37, 24, 31].

## III. BACKGROUND, RELATED WORK, & RESEARCH GAP

### A. *Related IoT Enabling Protocols & Monitoring Techniques*

Several solutions to solve the connection and cooperation of things under different exigences exist in IoT. Some permit long-range communication using low energy (*cf.* [34] for a survey on Low Power Wide Area Networks). Let us note as an example LORA [6], the counterpart of the solution is sporadic communication resulting in very low bandwidth (*cf.* [1] for the limitations of LORA). For these limitations, we exclude long-range networks from our study.

A significant step to creating a serviceable IoT domain is the adaptation of the functioning to IP protocols. IEEE 802.15.4 is a well-known and widely used standard launched in 2003. It defines how Low-Rate Wireless Personal Area Networks (LR-WPANs) operate and the specifications of the Medium Access Control (MAC) and Physical (PHY) layers for LR-WPANs [16]. Moreover, IEEE 802.15.4 can be used for many higher-layer standards, such as Zigbee, Wireless HART, and radio frequency for consumer electronics.

Intending to allow low-end devices with limited power to connect to the Internet, the IETF created 6LoWPAN in 2004 [26]. The goal of 6LoWPAN was to include an adaptation layer between the IPv6 and the IEEE networks. This layer has encapsulation and compression techniques to enable adequate IPv6 packets' transmission over IEEE 802.15.4 communication channels. For recent reviews and studies of 6LoWPAN *cf.* [9, 44, 12].

As mentioned above, one of the pertinent challenges in LLNs is to use an efficient routing protocol that meets the applications' requirements, such as considering low-power IoT devices and short transmission ranges. In response to these challenges, standardization groups, specifically the Internet Engineering Task Force (IETF) and the Institute of Electrical and Electronics Engineers (IEEE), standardized a broadly applied Routing Protocol for Low-Power and Lossy Networks (RPL) that was proposed in RFC 6550 [4].

In this cost-based routing protocol, a Destination-Oriented Directed Acyclic Graph (DODAG) is built, directed from the things to a central node corresponding to a border router (BR) toward the other parts of the Internet. In the DODAG, nodes are organized into a "layered" tree, starting from a single root (the BR). The construction of each tree-like DODAG is based on the attribution of" ranks" (the rank defines the position of the node in the DODAG to the neighbors and the BR descending from the root), which is computed via applying an Objective Function (OF) which can eventually be defined based on QoS metrics (e.g., delay). Usually, a node forwards the data to a parent (with a lower cost, *i.e.*, lower rank) toward the sink. The DODAG can be used to send messages to things and actuators [9].

Network monitoring tools generally aim at detecting and localizing network faults and taking corrective actions. A monitoring technique ensures earlier detection of failures and accelerates the repair. RPL proposes local repair and global *repair* mechanisms to reestablish the routing structure if it is failed [9]. These repair mechanisms are *reactive* since they are activated when a failure is detected. Consequently, nodes might be unreachable. As reported in [37], the average unreachable time of the node during DODAG reconstruction is almost three and a half minutes, which is the reason why numerous researchers have expressed concern over the routing issues in the IoT environment (*cf.* [3, 2]). For a recent review of the security of RPL-based 6LoWPAN networks in the Internet of Things [44].

A relatively similar problem is the monitoring of Wireless Sensor Networks, for which several monitor placement

TABLE II
CENTRALIZED MONITORING ARCHITECTURES' PROS & CONS

| Centralized monitoring | |
|---|---|
| Advantages: | • allows for simpler network management<br>• The base station is always assumed to be accessible and can be equipped with unlimited resources,<br>• can perform complex management tasks, thus, reducing the processing burden on resource-constrained nodes, and<br>• the base station has a global knowledge of the network, and therefore, it can provide accurate management decisions. |
| Disadvantages: | • it incurs a high message overhead (bandwidth and energy) from data polling, and this limits scalability,<br>• the base station is a single point of failure,<br>• they limit the possibility of creating ad-hoc domains without dedicated infrastructures,<br>• they represent a more static worldview, where device roles are fixed, rather than a dynamic worldview that recognizes that networks and devices, and their roles, may change over time, and<br>• if a network is partitioned, nodes that cannot reach the base are left without any management functionality. |

algorithms have been proposed. An extensive survey of MAC protocols can be found regarding mission-critical applications in [17]. There are a few propositions to supervise IoT network nodes [22, 23]. They propose passive monitoring techniques that use RPL in which the monitors are special, higher-order devices not limited in their resources. Unfortunately, this constraint is challenging since higher-order devices typically only constitute the minority of nodes in IoT networks. Consequently, it is only possible to cover the nodes and links partially.

The management of the monitoring system (and its scheduling) can be *centralized or distributed* (*cf.* [38, 25]). In centralized network management, a central entity, generally known as the base station, acts as the management station that collects information from all nodes and controls the entire network. Table II summarizes the advantages and disadvantages of centralized monitoring.

In previous work, energy-efficient monitor placement and scheduling in 6LowPAN were formulated in [27] as a multi-objective scheduling problem. The paper proposes a centralized computation where the objectives cover the minimization of energy consumption and the communication cost of monitoring.

The proposition is split into three phases. At first, the potential monitor sets (minimal vertex cover sets) are generated. The energy-efficient alternation of monitor sets needs the assignment of monitor sets on time periods. This subproblem is modeled as a Multi-Objective Generalized Assignment Problem. In the third phase, nodes' state transitions are optimized by solving a Traveling Salesman Path Problem (each node corresponding to a monitor set in a period). As a result of the decomposition, the method is *not exact* but gives optimums in each individual phase.

In subsequent work (*cf.* [28]), the exact formulation of the corresponding NP-hard optimization problem is described. The proposed model is based on a Binary Integer Program. The computed solution of centralized scheduling is optimal. However, as the problem size gets larger, the networks get denser. As a result, computing the assigned monitors' optimal schedule results requires a significantly long time. Hence, a distributed and simple mechanism is preferable. sporadic communication resulting in very low bandwidth (*cf.* [1] for the limitations of LORA). For these limitations, we exclude long-range networks from our study.

### *B. Research Gap & Contributions*

Compared to existing solutions, the roadblocks to overcome include integration and interoperability to standardized protocols and advanced technologies across the value chain (devices, networks, middleware, service platforms, and application functions) to foster smart, sustainable coverage of user needs for IoT services and products in the specific real-life scenarios of the pilot.

Concerning the potential application of 6LowPAN-based IoT networks and applying RPL in systems with a critical mission, our work focuses on developing a proactive monitoring solution that monitors *cover* the entire IoT network. From the point of view of efficiency, the *placement of monitors* is crucial. Consequently, the following research question is posed to ensure full monitoring coverage: How many monitors are required, and where should they be placed?".

Considering the limited computational capacity of simple IoT devices, it is imperative to *reduce and distribute* the added energy and communication cost of monitoring. To preserve the power of batteries and prolong the network's lifetime in battery-powered WSNs and IoT, *duty cycles* (alternations of awake and sleep states of nodes performing primary functions) are usual, required techniques [17].

A crucial requirement is achieving *real-time adaptability* to network changes by providing a dynamic worldview that recognizes that network connectivity and devices' health and roles may change over time. Consequently, monitoring coverage should be ensured while respecting the devices' *limited and changing resources*.

Moreover, given the fragility of the centralized solutions, and the controls, we propose a distributed and simple scheduling mechanism to compute and alternate monitor sets.

In a nutshell, this work proposes a *proactive, dynamic, and distributed* network monitoring mechanism with monitor placement and scheduling for 6LoWPAN-based IoT networks intended for mission-critical applications. The proposed mechanism aims to ensure real-time, efficient monitoring coverage while respecting devices' limited and changing power resources to prolong the network lifetime.

According to the comprehensive literature review performed in this research, to the best of our knowledge, no research work has proposed monitoring models with dynamic, energy-efficient role scheduling and integration with the standardized RPL and 6LoWPAN protocols.

## IV. PROPOSED PROACTIVE DYNAMIC IoT NETWORK MONITORING TECHNIQUE

### A. *IoT monitoring specifications, requirements, & objectives*

This section explains the main requirements, assumptions, and objectives for network monitoring for resource-constrained IoT.

One of the critical requirements is designing a monitoring mechanism that is entirely interoperable with the standardized IoT protocol suite, especially the IPv6 for Low-power Wireless Personal Area Networks (6LoWPAN) and the Routing Protocol for Low-power and lossy networks (RPL).

As mentioned in the Related Work section, the authors in [23] proposed a monitoring technique in which the monitors are special, higher-order devices with unlimited resources. Since higher-order devices typically only constitute the minority of nodes in IoT networks, the nodes, and links could only be partially covered, risking the possibility of an undetected node failure, which is unacceptable in mission-critical applications. For this reason, in the proposed model it is required to perform monitoring by ordinary, resource-constrained nodes in the application.

Moreover, we propose a passive monitoring technique to supervise the network's state and the availability of nodes and links. This requirement ensures observing the network's functioning and traffic without causing additional monitoring traffic and overhead.

Since monitoring is only one of the activities performed by the things, the power of batteries is consumed by monitoring and other activities like sensing, transmitting, and receiving. Such activities are the primary activities defined by the IoT network's original mission; the monitoring mechanism does not control them. However, the eventual changes in the power resources as a result of the primary function of the things should be dynamically observed since they affect the scheduling of the monitors. Table III summarizes the proposed model's requirements.

TABLE III
MONITORING SPECIFICATIONS & REQUIREMENTS

| |
|---|
| **Interoperability.** Monitoring should be interoperable with the standardized IoT protocol suite, specifically 6LoWPAN and RPL protocols. |
| **Passive Monitoring using resource-constrained nodes.** |
| **Efficient Monitoring.** Things have stringent *resource constraints* with only a fraction of the battery reserved for monitoring.<br>**Pervasive Monitoring.** In addition to the monitoring role, Things perform sensing, transmission, and/or actuation.<br>**Dynamic Monitoring.** Real-time adaptability that recognizes the change in network connectivity and devices' health. |

Given the above-stated requirements, **the objectives of the propositions of this work are as follows:**

- improving the resilience of critical-mission IoT domains via scalable, real-time monitoring that covers all network elements belonging to the concerned instance,
- balancing energy usage between monitors and following the eventual changes in the topology. The monitor set can (should) be changed dynamically, and
- computing (electing) the monitor set should be distributed.

be changed dynamically.

### B. *Overview of Controlled Greedy Sleep(CGS) Algorithm*

The Controlled Greedy Sleep (CGS) algorithm proposed in [40] targets Wireless Sensor Networks used to monitor an area. Leveraging the high redundancy feature usually present in sensor networks, the mechanism's objective is ensuring that a required number $k$ of sensors can provide measurements from each point in the area.

The mechanism is periodical; the lifetime of the WSN is prolonged by using different sensor sets in the periods. The selection and scheduling (duty-cycling) mechanism should ensure the $k$-coverage by the active sensors at each period. The distributed CGS provides a quasi-optimal sensor scheduling solution while respecting sensor node deployment and energy constraints. The same idea (duty-cycling) is used to organize network monitoring. Precisely, it is required that the additional monitoring load is distributed on the nodes by alternating between the monitor sets.)

The sensing assignment in the sensor network is represented by a bipartite graph $G_a = (S \cup R, E)$, where two disjoints sets of vertices represent the nodes $S$ and geographical regions $R$ (*cf.* Fig. 1), respectively. A region is the set of points in the area that a given sensor set can cover. In $G_a$, there is an edge $e$ between sensor $s \in S$ and region $r \in R$ if and only if $s$ covers region $r$. Sensors covering the same region can communicate directly since the communication range is at least twice the sensing range. The algorithm applies a *drowsiness factor*,

TABLE IV
GLOSSARY OF MODELING TERMS

| Term | Description |
|---|---|
| $k$ | Required number of sensors to provide measurements from each point in the monitored area. |
| $S$ | Set of vertices representing the sensing nodes, $s \in S$. |
| $R$ | Set of vertices representing the geographical regions, $r \in R$. |
| $G_a$ | Bipartite graph representing the sensing assignment in the sensor network, consisting of the two disjoint sets $S$ and $R$ and the set of edges $E$. |
| $E_s$ | Remaining energy of sensor node $s$ |
| $D_s$ | *Drowsiness factor* of sensor $s$, which represents the state of the sensor and its desire to sleep. |
| $\Phi_r$ | Coverage ratio of region $r$. |
| $C_r$ | The number of sensors covering region $r$ (the degree of $r$ in $G$). |
| $DTD_s$ | Decision Time Delay of node $s$, the time elapsed until each node $s$ decides whether to stay awake or go to sleep. |
| $AM$ | An Awake Message broadcast by $s$ to inform the other nodes of its decision to stay awake. |
| $DL_s$ | Delay List of node $s$ |
| $LAN_s$ | List of Awake Neighbors of node $s$. |

which models the state of the sensors and their "desire" to sleep. The factor is computed at the beginning of each period

for each node in a distributed manner. Supposing that a sensor node $s$ has $E_s$ remaining energy and can cover a set $R_s \subseteq R$, its drowsiness factor $D_s$ is defined as follows:

$$D_s = \left\{ \begin{array}{cc} \frac{1}{E_s^{\alpha}} \sum_{r \in R_s} \Phi_r & \text{if } \Phi_r > 0, \forall r \\ -1 & \text{otherwise.} \end{array} \right\} \quad (1)$$

where $\alpha$ is a positive constant (e.g. $\alpha = 2$), and $\Phi_r$ is the coverage ratio of region $r$, defined as follows:

$$\Phi_r = \left\{ \begin{array}{cc} \frac{1}{C_r - k} & \text{if } C_r > k \\ -1 & \text{otherwise.} \end{array} \right\} \quad (2)$$

Here, $C_r$ is the degree of the region $r$ in $G$, *i.e.*, the number of sensors covering $r$. $k$ is the desired level of redundancies in the coverage. This so-called "coverage ratio" $\Phi_r$ is positive if the region $r$ is over-covered, *i.e.*, more than $k$ sensors can cover it, and negative otherwise.

The drowsiness factor expresses a certain degree of the critical situation of the sensor. A sensor covering regions with low over-coverage could and should participate in more possible solutions than those covering regions also covered by many other sensors. The drowsiness factor is computed as the sum of the coverage ratios of the regions the sensor can observe. Consequently, sensors in critical positions could go to sleep whenever possible. Moreover, the drowsiness factor considers the energy of the sensor $s$; the smaller the sensor's energy, the larger its drowsiness. This factor permits a trade-off between energy usage and critical situations.

Depending on $D_s$, each node $s$ computes a Decision Time Delay $DTD_s$ inversely proportional to $D_s$ and broadcasts it to its neighbor. When a sensor decides to be awake, it informs the neighbor nodes with an awake message ($AM$). From the received $DTD$ and $AM$ messages, each node builds a Delay List ($DL_s$) and a List of Awake Neighbors ($LAN_s$). After $DTD_s$ time elapsed, each node $s$ decides based upon its lists: If all $r \in R_s$ can be covered using only nodes present in $LAN_s$ and nodes present in $DL_s$ (these latter nodes are nodes not yet decided), then node $s$ goes to sleep. Otherwise, $s$ decides to be active and broadcasts an $AM$ to inform the other nodes of its decision. Briefly, the CGS algorithm works as follows:

1) Run the network for a period of T
2) Wake up all sensors
3) Nodes with energy enough for at least one more period broadcast local Hello messages containing node geographical location
4) Each node $s$ calculates its own drowsiness factor $D_s$
5) Based on $D_s$ each node selects a Decision Time Delay ($DTD_s$)
6) Each node $s$ broadcasts its $DTD_s$ and collects other nodes' $DTD$ and $AM$
7) From the received messages and after $DTD_s$, each node $s$ decides its state for the next period.

This mechanism is a valid starting point for IoT network monitoring, but necessary adaptations are needed.

The following Section describes the proposed model and algorithmic solution.

Suppose a critical mission is realized using a DODAG topology $G_t = (V_t, E_t)$ for routing.

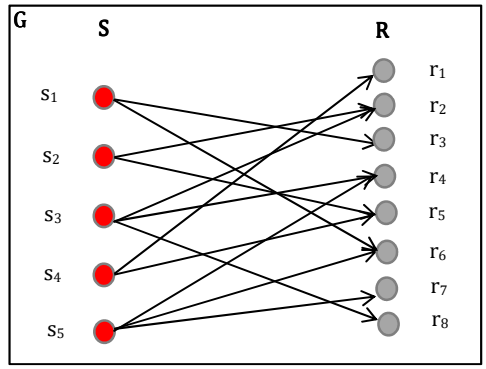



Figure 1. Bipartite graph showing the coverage of the sensors $(s_1, ..., s_5)$ and the sensing regions $(r_1, ..., r_8)$

### C. *Organization and Concepts*

The starting point of the developed proposition is based on the mechanism of the Controlled Greedy Sleep (CGS) [40] algorithm. Necessary adaptations are needed to satisfy the IoT network monitoring requirement.

**The proposition contains two major elements:**

- a cooperation protocol between nodes to assure the distributed scheduling, and
- an efficient computation algorithm to prepare the monitors' awake/sleep decisions.

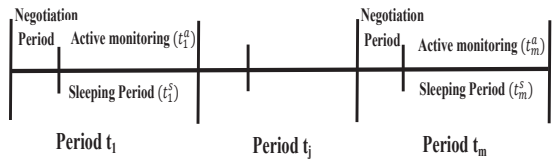



Figure 2. Monitoring Timeline, $T = \{t_1, t_2, ..., t_m\}$.

The monitoring activity is organized in a timeline that is decomposed into a sequence of periods, $T = \{t_1, t_2, ..., t_m\}$ (*cf.* Fig. 2). Each period is characterized by the set of active monitors and the duration of the period: $t_m = (S_j^a, t_j)$, where the set $S_j^a$ is the active monitor subset during $t_j$ that solves the coverage of the graph representing the *current* network topology. All periods assume the same period length. Moreover, a significant topology change (for instance new duty cycle) involves a new computation timeline T for the monitoring. The following definitions for node sets are used to describe the monitor selection and scheduling algorithm:

Consequently, it is essential to increase the knowledge of every node such that it knows the neighbors of its neighbors. One neighbor of one of its neighbors is usually called Neighbor-of-Neighbor ($NoN$) [20].

- *Awake Neighbors.* Similarly to CGS, each node $n$ and at each moment of negotiations should know the set of concurrent monitors that are still awake *for monitoring*. This subset of $S_n$ can be represented by the List of Awake Neighbors of $n$ $LAN_n$.

TABLE V
NODE SETS IN THE MONITOR SELECTION AND SCHEDULING ALGORITHM

| | Definition |
|---|---|
| - *Candidate neighbor set.* | Each node $n \in V_t$ has its candidate neighbor set $N_n$, which is the set of nodes in the reception range of $n$ (these nodes can be safely observed by $n$). It is the set of a potential target for $n$ to monitor (set $R_n$ in the bipartite graph $G$ defined by the CGS mechanism). |
| - *Concurrent monitor set.* | This set $S_n$ is composed of other potential concurrent monitors that can monitor at most one of the potential targets of $n$. Nodes in $N_n$ but not only them are in $S_n$. A node $x$, which is outside the reception range of $n$ but can monitor a node $y$ inside of $N_n$, should be considered a concurrent monitor node for $n$. $x$ is a neighbor node of $y$, a neighbor of $n$. |

Symmetric links are assumed; $n$ can communicate with the neighbor nodes in $N_n$ and observe them. During monitoring, a node selected as a monitor covers a subset of its candidate neighbor set.

Similar to CGS, at the beginning of each monitoring period, there is a short negotiation period (*cf.* Fig. 2), where the communication between neighbors is established. This periodical communication between neighboring nodes must be accomplished for the following purposes:

- updating the candidate neighbor set $S_n$ for all $n \in V_t$, so that the current List of Awake Neighbors $LAN_n$ is known, which constantly changes due to the lossy nature of 6LoWPANs or simply because of the applied duty cycling mechanism,
- informing neighboring nodes of the updated coverage ratio of each node, and
- informing neighboring nodes of each node's monitoring awake/sleep decision for the next period.

The problem presented in this work is mapped to the one dealt with by the CGS algorithm [40], albeit with significant differences.

- In CGS, the set $S$ of concurrent sensors and observed regions $R$ are disjoint (they are geographical regions, *cf.* Fig. 1). The two sets are composed of the same nodes for the problem in hand, where a *monitored* node can also be a monitor for another. The organization of the node sets is not a simple duplication of nodes. The edges should reflect the real possibilities of monitoring (the NoN set of nodes should be considered).
- In CGS, sensors are identified by their geographical locations. On the other hand, in 6LoWPAN-based IoT networks, the nodes' radio communication ranges are defined by link-local reachability, where nodes are discovered by the 6LoWPAN *Neighbor Discovery Protocol (NDP)* [1], and identified by unique RIME/IPv6 addresses.
- There is an edge between monitor $s$ and element $r$ in the graph $G_a$, if and only if $r$ is within the radio environment of $s$. For monitor placement, the direction of edges is irrelevant, which implies that if there is a directed edge from $s$ to $r$, $s$ will be able to monitor $r$, and $r$ can monitor $s$. Undirected graphs are used in several routing protocols, as in the models of [36], and [10].
- In the area coverage by WSNs, sensors that can observe the same region are neighbors for communications, a fact that does not apply to monitoring the wireless network itself. In the proposed dynamic monitor scheduling algorithm, it is essential to increase the knowledge of every node, such that it knows the state of all nodes that can observe at least one of its neighbors. Moreover, each node must know the neighbors of its neighbors, known as the Neighbor-of-Neighbor ($NoN$) set [20]. A node is awake/sleep schedule in the next period $t_j$ is affected by the state of its $NoN$. The concurrent monitor set of a node $n$ is the union of its neighbor and $NoN$ sets, as given in (3).

$$S_n = N_n \bigcup NoN_n \tag{3}$$

The requirement of the knowledge of $NoN$s is illustrated in Fig. 3.

For $v_1$ and $v_3$, $N_1 = \{2, 3, 5\}$, and $N_3 = \{1, 6\}$ respectively. Suppose that it is decided that $v_6$ is "sleep-monitoring" in the next period $t_j$. If $v_1$ goes to sleep mode in the same period, $v_3$ will not be covered. Therefore, $v_1$ should know its neighbors of neighbors, which includes $NoN_1 = \{v_4, v_5, v_6, v_7\}$. For $v_1$, knowing that a member of its $NoN_1$, namely, $v_6$, is sleeping, it *should* decide to stay "active-monitoring". Otherwise, the neighbor of $v_1$, $v_3$, will not be covered.

The bipartite graph giving the relations between this illustrating network's potential monitoring and monitored nodes and a possible monitoring set of nodes are depicted in Fig. 4.

In our proposition, the monitoring system is relatively simple (in this case, the minimal coverage ratio $k$ is equal to 1). It is a cheap and straightforward mechanism. The inconvenience of this solution is that some nodes (eventually monitor-actives) can be in a critical situation in the monitoring. It is the case when an active node is the only one monitoring another node. This case is illustrated in Fig. 4. In this example, nodes 2, 3, and 6 are monitored by only one monitor. Suppose that node 3 (which is also a monitor) fails. Then, until the repair of this failure, node 6 is not monitored. A $k$-coverage of nodes with $k > 1$ constraint can be applied to improve the fault tolerance of the monitoring system. For instance, by applying a 2-coverage of nodes, the monitoring system will tolerate a first failure of a monitor and can continue the monitoring. However, a node failure impacts the network's primary critical mission and the communication between the nodes and the BR. Consequently, after detecting a failure, it is necessary to repair the DODAG used by the application and immediately recompute its monitoring system.

### *D. Scheduling Mechanism*

The proposed monitoring and the corresponding scheduling are described in Algorithms 1, 2, 3, and 4. The monitoring

[1]NDP is a messaging protocol that facilitates the discovery of neighboring devices over a network [39].

must function during the timeline's length, represented by $Timeline\_Length$. At the beginning of a new monitoring period, $t_m$, all nodes wake up (Algorithm 2 Step 2.1), estimate their remaining power ($E_s$), and initialize their parameters. Nodes with a remaining energy level high enough for monitoring (more than a given $Energy\_Threshold$) for at least one more period locally broadcast an Awake Message ($AM$) (Step 2.8). Otherwise, to conserve the remaining power for its primary function (sensing, actuation, and transmission), it broadcasts a Sleep Message $SM$ and chooses the "monitoring-sleep" state (Steps 2.9 & 2.10 ). Naturally, those nodes are considered "sleeping" nodes.

It is noteworthy that the monitoring mechanism does not influence the node's primary duty cycle, *i.e.*, the radio is turned off only if it is idle for its primary function. Our computation concerns only the monitoring task, and the decisions are to select either a state of "monitoring-active" or a state of "monitoring-sleep".

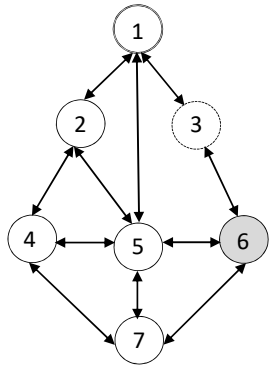



Figure 3. For $v_1$, knowing that a member of its $NoN_1$, namely, $v_6$, is sleeping decides to stay active-monitoring to ensure that its neighbor, $v_3$, is covered.

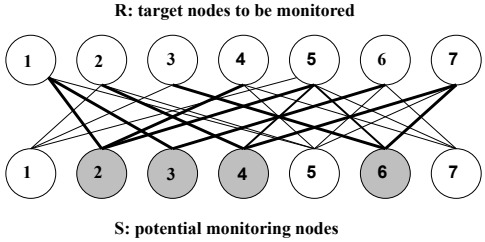



Figure 4. A possible result of the election of monitors in the example of Fig. 3.

When a node $n$ receives a message from a neighbor, there are several tasks to perform: (1) update its List of Awake Neighbors ($LAN_n$), either by adding or removing this neighbor's address according to the neighbor's received state (monitoring-active or monitoring-sleep) (Algorithm 3 Step 3.2). Then, (2) update its list of Neighbors-of-Neighbors ($NoN_n$) from the received list of neighbors, $LAN_{neighbor}$ (Step 3.3). (3) Compute its own coverage ratio (*cf.* Equation 4), and (4) update its Delay List $DL_n$ (Steps 3.4 & 3.5). Finally, (5) broadcast the updated parameters to its neighbors (Step 3.8).

At the end of the negotiation period, each node has to decide whether it will be "monitoring-active" or "monitoring-sleep" for the rest of the monitoring period. Algorithm 4 describes how these decisions are made, which mainly depend on the $drowsiness\_factor_n$ (*cf.* Equation 1). Each monitor's drowsiness factor includes the sum of coverage ratios of the objects it can monitor. Negative drowsiness indicates that a monitor cannot choose the "monitor-sleep" state. The smaller the energy $E_n$ of a monitor candidate, the larger its drowsiness factor. On the contrary, minor drowsiness means a long Decision Time Delay ($DTD$) (*cf.* Equation 5). These delays provide priorities when nodes announce their Awake Messages ($AM$).

A monitor participating in several critical coverages is more likely to engage in several possible solutions than other potential monitors simultaneously covered by alternative nodes. Therefore, they have more significant drowsiness factors (*cf.* Equation 4). This property forces the nodes in critical situations to deactivate monitoring whenever possible and permits the loading of monitors which are in less critical situations.

Each node $n \in V_t$ has received the $coverage\_ratio_{neighbor}$ of the members of its $LAN_n$ and $NoN_n$. If at least one of its neighbors or Neighbors-of-Neighbors is under-covered, (*i.e.*, has a negative coverage ratio), this indicates that at most one node can monitor it. Therefore $n$ decides to stay awake for monitoring to maintain successful coverage (Step 4.9). Accordingly, it broadcasts an $AM$ (Step 4.10). Otherwise, it *can* choose the "monitoring-sleep" decision depending on the comparison between its own $DTD_n$ (*cf.* Equation 5) with its neighbors' $DTD_{neighbor}$. The different $DTD_{neighbor}$ values were previously received and saved in the Delay List $DL$ (Step 4.4). In the case where $n$ has the smallest $DTD_n$, it broadcasts an $SM$ and turns off the monitoring activity (Steps 4.5 - 4.7).

$$\Phi_r = \left\{ \begin{array}{cc} \frac{1}{C_r - 1} & \text{if } C_r > 1 \\ -1 & \text{otherwise.} \end{array} \right\} \tag{4}$$

$$DTD = \left\{ \begin{array}{cc} \frac{1}{D_s} & \text{if } D_s > 1 \\ 0 & \text{otherwise.} \end{array} \right\} \tag{5}$$

The decision delays $DTD$ of nodes are less than the length of the negotiation period. In this manner, all available nodes in the network decide to be monitor-active or monitor-sleep before the next period. As it was shown, if the concurrent and awake monitoring nodes of a node $n$ can ensure the coverage of the potential target nodes of $n$, then this latter chooses the monitor-sleep state.

*Property 1:* In each period, the set of monitor-active nodes of a connected DODAG is a minimal covering set[2].

## V. Experimentation of the Proposition

The following experimentation illustrates the functioning and the performance of distributed scheduling. We analyze the effect of the reserved battery level for monitoring, the period length on the distribution of energy usage, and the network size. There is no existing, similar heuristic to compare

[2]Remember, a minimal covering set is not obligatory a minimum set, but it can not be reduced without the loss of coverage.

**Algorithm 1** PROCEDURE DYNAMIC_DISTRIBUTED_MONITORING

**Input:** $Energy\_Threshold, Timeline\_Length, Period\_Length, Negotiation\_Period$
**Output:** Real-time monitoring schedule of 6LoWPAN-based IoT network

```
begin
1.1   while timeline_timer < Timeline_Length do
1.2     while period_timer < Period_Length do
1.3       while negotiation_timer < Negotiation_Period do
1.4         forEach n ∈ G do
1.5           START_UP();
1.6           RECEIVE_MESSAGE();
1.7           DECIDE_STATE();
1.8         end forEach
1.9       end while
1.10    end while
1.11  end while
end
```

**Algorithm 2** PROCEDURE START_UP

**Input:** $Energy\_Threshold$
**Output:** Initialize node state

```
begin
2.1   RADIO_ON();
2.2   if E_n > Energy_Threshold do
2.3     state_n ⟵ 1;
2.4     drowsiness_factor_n ⟵ -1;
2.5     coverage_ratio_n ⟵ -1;
2.6     DTD_n ⟵ 0;
2.7     LOCAL_BROADCAST(AM, state_n, drowsiness_factor_n, coverage_ratio_n, DTD_n);
2.8   else
2.9     LOCAL_BROADCAST(SM, state_n ⟵ 0);
2.10    RADIO_OFF();
2.11  end if
end
```

**Algorithm 3** PROCEDURE RECEIVE_MESSAGE

**Input:** Address of $neighbor$, $state\_neighbor$, $LAN\_neighbor$, $coverage\_ratio\_neighbor$, $DTD\_neighbor$
**Output:** Update $LAN_n$, $NoN_n$, $coverage\_ratio_n$, $DL_n$; LOCAL_BROADCAST updated parameters

```
begin
3.1   if state_neighbor > 1 do
3.2     LAN_n ⟵ LAN_n ∪ neighbor;
3.3     NoN_n ← NoN_n ∪ LAN_neighbor;
3.4     UPDATE_COVERAGE_RATIO(coverage_ratio_neighbor);
3.5     UPDATE_DL(DTD_neighbor);
3.6   else LAN_n ⟵ LAN_n/neighbor;
3.7   end if
3.8   LOCAL_BROADCAST(state_n, coverage_ratio_n, LAN_n, NoN_n);
end
```

**Algorithm 4** PROCEDURE DECIDE_STATE

**Input:** $LAN_n$, $NoN_n$, $DL_n$

**Output:** node $s$ decides whether to stay active monitoring or sleep in $t_j$ and accordingly broadcast $AM$ or $SM$

**begin**

4.1 **if** $coverage\_ratio_{neighbor}$ & $coverage\_ratio_{NoN} >= 0$ $\forall neighbor \in LAN_n, \forall NoN \in NoN_n$ **do**

4.2 COMPUTE_DROWSINESS_FACTOR();

4.3 COMPUTE_DTD();

4.4 **if** $DTD_n < DTD_{neighbor}$ $\forall neighbor \in LAN_n$, $\forall DTD_{neighbor} \in DL_n$

4.5 LOCAL_BROADCAST($SM, state_n \longleftarrow -1$);

4.6 WAIT($DTD_n$);

4.7 RADIO_OFF();

4.8 **end if**

4.9 $drowsiness\_factor_n \longleftarrow -1$;

4.10 LOCAL_BROADCAST($AM$);

4.11 **end if**

**end**

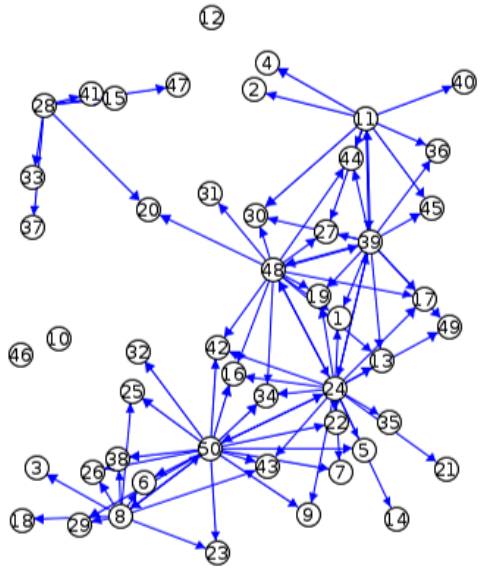


Figure 5. Radio communication within a network of 50 devices of type WisMote; node 1 is the Border Router.

with our distributed solution. The proposition for IoT network monitoring in [23] is different because it is based on particular monitor nodes.

### A. *Implementation & Experimental Setup*

The proposed monitoring system for the resilience of critical IoT domains is implemented on the Contiki Operating System. For dynamic monitoring placement and scheduling, nodes' current power levels should be estimated as accurately as possible. The device's power is allocated to monitoring and, more importantly, to the primary activities, including sensing, actuation, processing, and transmission. The monitoring schedule should be efficient enough not to influence the energy

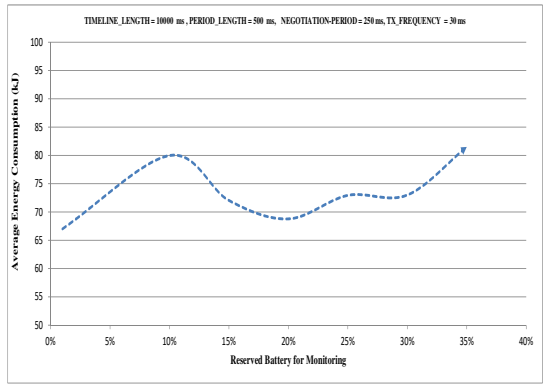



Figure 6. Effect of varying the size of the reserved battery for monitoring on the average energy consumption. Timeline Length = 10000 ms, Negotiation Period = 50 $ms$, $Tx$ Frequency = 30 ms.

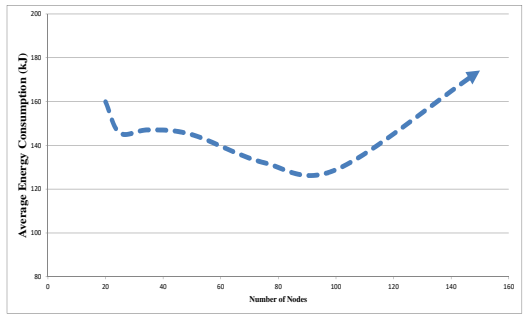
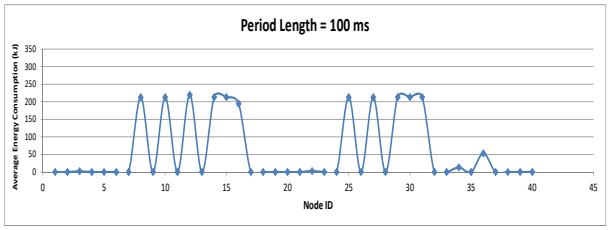
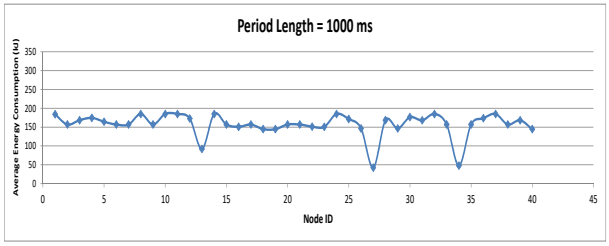


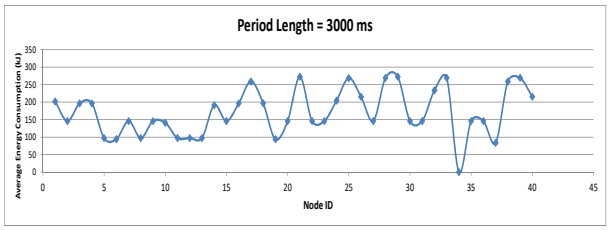

Figure 7. The average energy consumption of different-sized networks, Timeline Length = 10000 ms, Period Length = 2000 ms, Negotiation Period = 50 ms, Tx Frequency = 30 ms, Reserved Battery = 10%.




(a)




(b)




(c)

Figure 8. Effect of varying the period length on the average energy consumption, (a) period length = 100 *ms*; (b) period length = 1000 *ms*; (c) period length = 3000 *ms*. Timeline Length = 10000 ms, Negotiation Period = 50 ms, Tx Frequency = 30 ms, Reserved Battery = 10%

required to perform the primary functions. The WisMote [5] is taken as a candidate platform for the monitoring mechanism. It features a 16-bit MSP430 with 20-bit support, 16 $kB$ RAM, a nominal 128 $kB$, 192 $kB$ or 256 $kB$ ROM, and CC2520 radio transceiver, with light, battery, and radio sensors. It is powered by a pair of AAA batteries with 3 volts. The total energy available by the WisMote is calculated as follows:

$$2 \times (1.15\ Ah) \times (1.5\ V) \times (3600\ s) = 11421\ J = 11421000\ mJ \tag{6}$$

The `POWER_TRACE` procedure is embedded in Contiki to estimate the current energy level of nodes. `POWER_TRACE` procedure embedded in Contiki is used. Its output is printed in timer ticks as follows:

- $tx$ - the number of ticks the radio has been in transmit mode *(energest-type-transmit)*
- $rx$ - the number of ticks the radio has been in receive mode *(energest-type-listen)*
- $cpu$ - the number of ticks the CPU has been in active mode *(energest-type-cpu)*
- $cpu - idle$ - the number of ticks the CPU has been in idle mode *(energest-type-lpm)*

With each call of the `START_UP` procedure, `POWER_TRACE` is called, and the current energy level $E_s$ is estimated by executing the following computations (Algorithm 2 Step 2.2).

$$\textit{ticks-in-tx-mode} = \textit{energest-type-time} \times \textit{energest-type-transmit} \tag{7}$$

$$\textit{seconds-in-tx-mode} = \frac{\textit{ticks-in-tx-mode}}{\textit{rtimer-arch-second}} \tag{8}$$

To compute the average current consumption (in milliamperes, $mA$), multiply each of $tx, rx, cpu$, cpu-idle with the respective current consumption in that mode in $mA$ (the values are obtained from the datasheet of the node), sum them up, and divide by *rtimer-arch-second*,

$$\textit{current} = tx \times \textit{current-tx-mode} + rx \times \textit{current-rx-mode} + cpu \times \textit{current-cpu} + \frac{\textit{cpu-idle} \times \textit{current-idle}}{\textit{rtimer-arch-second}} \tag{9}$$

$$\textit{charge} = \frac{\textit{current} \times (cpu + \textit{cpu-idle})}{\textit{rtimer-arch-second}} \tag{10}$$

To compute the power (in milliwatts, $mW$), multiply the average current consumption by the voltage of the device:

$$\textit{power} = \textit{current} \times \textit{voltage} \tag{11}$$

Finally, to compute the energy consumption (in millijoules, $mJ$), multiply the power with the duration in seconds or multiply the charge with the voltage of the system:

$$\textit{energy} = \textit{charge} \times \textit{voltage} \tag{12}$$

### B. *Experimental Results*

Experimentation is performed within the Contiki OS using the COOJA network simulator, the *de facto* simulator for constrained-IoT applications. The dynamic distributed monitoring mechanism is tested using network instances of random sizes and topologies (network size ranges from 20 to 200 nodes). Fig. 5 illustrates the radio communication and states in a network of 50 nodes of type WisMote. At the moment of the snapshot, only a subset of nodes is active for the communications (nodes 8, 11, 13, 24, 28, 39, 49, and 50). One can see the neighbors of these nodes, which receive (and can detect) the messages. Other nodes (nodes 10, 12, and 46) do not receive any messages and can not monitor the mentioned active node set; however, they can eventually monitor other nodes.

It is assumed that each node has a reserved battery for the monitoring activity across the entire timeline length, apart from the energy dedicated to the main functions. We tested the model's sensitivity towards variations in the reserved battery for monitoring during experimentation. Eight trials were run for which the reserved battery ranged from 1% to 35% of the total available battery of the WisMote, which corresponded to 112.10 - 3997.35 $kJ$. The $Timeline\_Length$, $Period\_Length$, $Negotiation\_Period$, and the frequency of transmission ($Tx-frequency$) were set in these trials to 10000

*ms*, 500 *ms*, 250 *ms*, and 30 *ms*, respectively. The results are displayed in Table VI.

Comparing the two extreme thresholds, one where the reserved battery is tightened the most (1%) and another where it is stretched to 35%, produced an interesting result: the average energy consumption in the case of the 1% reserved battery is reduced by 21.55%. This result highlights the model's adaptability towards tight energy constraints, as it strives to preserve scarce resources by effectively distributing the monitoring role. Some nodes decided not to participate in the monitoring activity, thus rendering a zero energy consumption level. Those nodes decided after ensuring that other monitors covered the entire set of neighbors.

Another set of experiments was designed to test the correlation between the period length and the average energy consumption. The *period_length* should be carefully chosen such that it is neither too short nor too long. A too-short *period_length* may result in false alarms. On the other hand, a shorter *period_length* may unnecessarily exhaust the energy of monitors as they are awake-monitoring for quite a long time. Each negotiation also corresponds to an additional cost, communications, frequent transitions, and an overhead for monitoring. A too-short *period_length* can lead to an unnecessary increase in overheads. A too-long *period_length* may drain some monitors' power as they are awake-monitoring for quite a long time, giving an unbalanced energy usage across the set of nodes.

Table VII displays the average energy consumption and the standard deviation in response to varying the *Period_Length*. The *Timeline_Length*, *Negotiation_Period*, and $Tx - frequency$ were fixed in all trials to 10000 *ms*, 250 *ms*, and 30 *ms*. A subset of those trials is displayed in Fig. 8. There is a trade-off between energy consumption and the balance of the monitoring load among the nodes. It can be seen in Fig. 8 (also *cf.* the Standard Deviation column in VII) that a very short *Period_Length*, 100 *ms*, results in an unfair distribution of the monitoring load, where some nodes exhaust comparatively high amounts of energy for monitoring while others are at a zero level consumption; which is illustrated by the high standard deviation value in Table VII. On the other hand, a too-long *Period_Length* of 3000 *ms* revealed a significant rise in the average energy consumption, which is justified by the long monitoring duty cycles. It is detected that the best combination of relatively low average energy consumption and a good balance between the monitoring loads is achieved when the *Period_Length* is set to 1000 *ms*.

The final set of experiments was performed to test the effect of the network size on energy consumption and the model's scalability. The network size was increased to 150 nodes and 3200 links. It can be seen from VIII that the proposition developed in this research is robust towards the network size. Results show that the percentage of energy consumption from the total available battery of WisMote never exceeds 1.36%, regardless of the network size. Fig. 7 depicts the increase in the average energy consumption against the network size, which is almost negligible.

TABLE VI
ENERGY CONSUMPTION OF A NETWORK OF 20 NODES WITH DIFFERENT LEVELS OF RESERVED BATTERY FOR MONITORING. $Timeline\_Length$ = 10000 *ms*, $Period\_Length$ = 500 *ms*, $Negotiation\_Period$ = 250 *ms*, $Tx - frequency$ = 30 *ms*.

| Reserved battery(%) | Reserved battery($kJ$) | Avg. consump.($kJ$) |
|---|---|---|
| 1 | 114.21 | 67.01 |
| 10 | 1142.10 | 79.97 |
| 15 | 1713.15 | 72.03 |
| 20 | 2284.20 | 68.78 |
| 25 | 2855.25 | 72.94 |
| 30 | 3426.30 | 72.99 |
| 35 | 3997.35 | 81.45 |

TABLE VII
ENERGY CONSUMPTION OF A NETWORK OF 40 NODES WITH DIFFERENT PERIOD LENGTHS. $Timeline\_Length$ = 10000 *ms*, $Negotiation\_Period$ = 50 *ms*, $Tx - frequency$ = 30 *ms*, RESERVED BATTERY = 10% (1142.1 KJ).

| Period Length (*ms*) | Avg. consumption ($kJ$) | Standard Dev. ($kJ$) |
|---|---|---|
| 100 | 60.17 | 95.23 |
| 500 | 142.27 | 144.20 |
| 1000 | 156.82 | 31.34 |
| 2000 | 94.64 | 105.65 |
| 2500 | 125.79 | 45.37 |
| 3000 | 215.43 | 66.41 |

TABLE VIII
AVERAGE AND PERCENTAGE OF ENERGY CONSUMPTION OF DIFFERENT-SIZED NETWORKS. $Timeline\_Length$ = 10000 *ms*, $Period\_Length$ = 500 *ms*, $Negotiation\_Period$ = 250 *ms*, $Tx - frequency$ = 30 *ms*.

| Number of nodes | Avg. consumption ($kJ$) | % of consumption |
|---|---|---|
| 20 | 159.89 | 1.36 |
| 25 | 145.61 | 1.27 |
| 50 | 144.89 | 1.26 |
| 75 | 132.09 | 1.27 |
| 100 | 128.89 | 1.14 |
| 150 | 174.21 | 1.13 |

## VI. CONCLUSION

The proposed model targets the dynamic distributed monitoring placement and scheduling of mission-critical IoT networks, with complete interoperability with the IoT standardized protocols. The model's dynamic feature ensures the real-time adaptation of the monitoring schedule to the frequent network instabilities without requiring to re-solve the monitoring placement and scheduling problems with each abrupt change in the network topology or the nodes' availability. The distributed feature aims to reduce the communication overhead between monitors and the Border Router, often resulting from centralized monitoring mechanisms.

The dynamic monitoring mechanism follows the basic idea of the Controlled Greedy Sleeping (CGS) algorithm proposed in [40]. Necessary adaptations on CGS for the scheduling of monitoring activities have been proposed. The monitoring awake/sleep schedule of nodes is computed using the notions of coverage ratio and drowsiness factor, which ensure the coverage of the entire set of critical nodes while prioritizing the awake/sleep decision based on coverage and energy levels. Successful neighbor discovery and knowledge about the

neighbors' state are achieved by inter-communication between nodes. This communication is scheduled at the beginning of each period, namely, the negotiation period. Nodes with critical monitoring coverage, *i.e.*, monitoring neighbors not covered by other monitors) are not allowed to sleep.

Performance evaluations and accurate energy levels estimation are achieved using Contiki/COOJA, the *de facto* network simulator for constrained IoT. Simulations were performed to evaluate the model's adaptability to tight energy constraints. The results show that the tighter the energy constraint, the lower the average energy consumption while ensuring full monitor-network coverage. The monitoring schedule guarantees a smooth operation of the things' main functions, as it strives to preserve scarce resources by effectively distributing the monitoring role. A sensitivity analysis was conducted within the experiments to obtain the "best" combination between the parameters and minimize the trade-off between them.

Compared to the previously proposed three-phase decomposition [40], the dynamic distributed heuristic achieves better computational complexity and scalability results. The only limitation is that the schedule is not *exact*. However, with the benefit of achieving robust, real-time adaptability to network changes and the distributed mechanism's reduced computational and communication overhead, the dynamic model's performance is superior. Further experimentation and comparisons between the two models are required regarding energy consumption, the size of the monitoring sets, and the time required to obtain the monitoring schedule, depending on the network's size. The evaluation of the dynamic heuristic's approximation factor is left for future work.

## REFERENCES

[1] F. Adelantado et al. "Understanding the Limits of LoRaWAN". In: *IEEE Communications Magazine* 55.9 (2017), pp. 34–40. https://doi.org/10.1109/MCOM.2017.1600613.

[2] David Airehrour, Jairo Gutierrez, and Sayan Kumar Ray. "Secure Routing for Internet of Things: A Survey". In: *Journal of Network and Computer Applications 66* (2016), pp. 198–213. ISSN: 1084-8045. DOI: 10.1016/j.jnca.2016.03.006. URL: https://www.sciencedirect.com/science/article/pii/S1084804516300133.

[3] David Airehrour, Jairo A. Gutierrez, and Sayan Kumar Ray. "SecTrust-RPL: A Secure Trust-Aware RPL Routing Protocol for Internet of Things". In: *Future Generation Computer Systems 93* (2019), pp. 860–876. ISSN: 0167-739X. DOI: 10.1016/j.future.2018.03.021. URL: https://www.sciencedirect.com/science/article/pii/ S0167739X17306581.

[4] Roger Alexander et al. *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*. RFC 6550. Mar. 2012. DOI: 10.17487/RFC6550. URL: https://rfc-editor.org/rfc/rfc6550.txt.

[5] ARAGO. *WisMote*. URL: https://github.com/contiki-os/contiki/tree/master/platform/wismote

[6] Martin Bor, John Vidler, and Utz Roedig. "LoRa for the Internet of Things". In: *Proceedings of the 2016 International Conference on Embedded Wireless Systems and Networks*. *EWSN* '16. Graz, Austria: Junction Publishing, 2016, pp. 361–366. ISBN: 9780994988607.

[7] Brian Buntz. *The Top 20 Industrial IoT Applications*. https://www.iotworldtoday.com/2017/09/20/top-20-industrial-iot-applications/. Accessed: 2020-11-08. 2017.

[8] Madhavi Latha Challa, K. L. S. Soujanya, and C. D. Amulya. "Remote Monitoring and Maintenance of Patients via IoT Healthcare Security and Interoperability Approach". In: *Cybernetics, Cognition and Machine Learning Applications*. Ed. by Vinit Kumar Gunjan et al. Singapore: Springer Singapore, 2020, pp. 235–245. ISBN: 978-981-15-1632-0. DOI: 10.1007/978-981-15-1632-0_22.

[9] Khalid A. Darabkh et al. "RPL Routing Protocol over IoT: A Comprehensive Survey, Recent Advances, Insights, Bibliometric Analysis, Recommendations, and Future Directions". In: *Journal of Network and Computer Applications 207* (2022), p. 103476. ISSN: 1084-8045. DOI: 10.1016/j.jnca.2022.103476. URL: https://www.sciencedirect.com/science/article/pii/S1084804522001242.

[10] Pierre Fraigniaud and George Giakkoupis. "Greedy Routing in Small-World Networks with Power-Law Degrees". In: *Distributed computing 27.4* (2014), pp. 231–253. DOI: 10.1007/s00446-014-0210-y.

[11] Peter Friess and Rolf Riemenschneider. "IoT Ecosystems Implementing Smart Technologies to Drive Innovation for Future Growth and Development". In: *Digitising the Industry Internet of Things Connecting the Physical, Digital and VirtualWorlds*. River Publishers, 2022, pp. 5–13. DOI: 10.1201/9781003337966.

[12] Matheus Araujo Gava et al. "Optimizing Resources and Increasing the Coverage of Internet-of-Things (IoT) Networks: An Approach Based on LoRaWAN". In: *Sensors 23.3* (2023). ISSN: 1424-8220. DOI: 10.3390/s23031239.

[13] Gopal Ghosh, Monica Sood, Sahil Verma, et al. "Internet of Things based video Surveillance Systems for Security Applications". In: *Journal of Computational and Theoretical Nanoscience 17.6* (2020), pp. 2582–2588. DOI: 10.1166/jctn.2020.8933.

[14] Brij B Gupta and Megha Quamara. "An overview of Internet of Things (IoT): Architectural aspects, challenges, and protocols". In: *Concurrency and Computation: Practice and Experience 32.21* (2020), e4946.

[15] Baofeng Ji et al. "Survey on the Internet of Vehicles: Network Architectures and Applications". In: *IEEE Communications Standards Magazine 4.1* (2020), pp. 34–41. DOI: 10.1109/MCOMSTD.001.1900053.

[16] Wafa'a Kassab and Khalid A Darabkh. "A–Z survey of Internet of Things: Architectures, protocols, applications, recent advances, future directions and recommendations". In: *Journal of Network and Computer Applications 163* (2020), p. 102663. DOI: 10.1109/ICCCNT.2017.8203943.

[17] Adam Kozłowski and Janusz Sosnowski. "Energy efficiency trade-off between duty-cycling and wake-up radio techniques in IoT networks". In: *Wireless Personal Communications 107.4* (2019), pp. 1951–1971. DOI: 10.1007/s11277-019-06368-0.

[18] In Lee. "The Internet of Things for enterprises: An ecosystem, architecture, and IoT service business model". In: *Internet of Things 7* (2019), p. 100078. ISSN: 2542-6605. DOI: 10.1016/j.iot.2019.100078.

[19] Ankur Lohachab and Bidhan Karambir. "Critical analysis of DDoS—An emerging security threat over IoT networks". In: *Journal of Communications and Information Networks 3* (2018), pp. 57–78. DOI: 10.1007/s41650-018-0022-5.

[20] Gurmeet Singh Manku, Moni Naor, and Udi Wieder. "Know Thy Neighbor's Neighbor: The Power of Lookahead in Randomized P2P Networks". In: *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. ACM. 2004, pp. 54–63. DOI: 10.1145/1007352.1007368.

[21] George Matta et al. "Risk management and standard compliance for cyber-physical systems of systems". In: *Infocommunications Journal 13.2* (2021), pp. 32–39. DOI: 10.36244/ICJ.2021.2.5.

[22] Anthéa Mayzaud, Rémi Badonnel, and Isabelle Chrisment. "A distributed monitoring strategy for detecting version number attacks in RPL-based networks". In: *IEEE Transactions on Network and Service Management 14.2* (2017), pp. 472–486. DOI: 10.1109/TNSM.2017.2705290.

[23] Anthéa Mayzaud et al. "Using the RPL Protocol for Supporting Passive Monitoring in the Internet of Things". In: *Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP*. IEEE. 2016, pp. 366–374. DOI: 10.1109/NOMS.2016.7502833.

[24] Bacem Mbarek, Mouzhi Ge, and Tomáš Pitner. "Proactive trust classification for detection of replication attacks in 6LoWPAN-based IoT". In: *Internet of Things 16* (2021), p. 100442. DOI: 10.1016/j.iot.2021.100442.

[25] Jozef Mocnej et al. "Decentralised IoT Architecture for Efficient Resources Utilisation". In: *IFAC-PapersOnLine 51.6* (2018). 15th IFAC Conference on Programmable Devices and Embedded Systems PDeS 2018, pp. 168–173. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2018.07.148. URL: http://www.sciencedirect.com/science/article/pii/S2405896318308942.

[26] Guido Moritz et al. "Beyond 6LoWPAN: Web services in wireless sensor networks". In: *IEEE Transactions on Industrial Informatics vol.9* (Nov. 2013), pp.1795, 1805. **DOI**: 10.1109/TII.2012.2198660.

[27] B. Mostafa et al. "An Energy-Efficient Multiobjective Scheduling Model for Monitoring in Internet of Things". In: *IEEE Internet of Things Journal 5.3* (2018), pp. 1727–1738. ISSN: 2327-4662. **DOI**: 10.1109/JIOT.2018.2792326.

[28] Basma Mostafa et al. "Optimal proactive monitor placement & scheduling for IoT networks". In: *Journal of the Operational Research Society 73.11* (2022), pp. 2431–2450. **DOI**: 10.1080/01605682.2021.1992310.

[29] Radouan Ait Mouha. "Internet of Things (IoT)". In: *Journal of Data Analysis and Information Processing 9.2* (2021), pp. 77–101. **DOI**: 10.4236/jdaip.2021.92006.

[30] Seyyed Esmaeil Najafi, Hamed Nozari, and Seyyed Ahmad Edalatpanah. "Investigating the Key Parameters Affecting Sustainable IoT-Based Marketing". In: *Computational Intelligence Methodologies Applied to Sustainable Development Goals*. Ed. by José Luis Verdegay, Julio Brito, and Carlos Cruz. Cham: Springer International Publishing, 2022, pp. 51–61. ISBN: 978-3-030-97344-5. **DOI**: 10.1007/978-3-030-97344-5_4.

[31] Nasser Al-Qadami and Andrey Koucheryavy. "Fault-Tolerance Algorithm in Wireless Sensor Networks." In: *Infocommunications Journal*, Hungary, ISSN (2015), pp. 2061–2079.

[32] Rixuan Qiu et al. "A Fine-grained Dynamic Access Control Method for Power IoT Based on Kformer". In: *Infocommunications Journal 14.4* (2022), pp. 79–85. **DOI**: 10.36244/ICJ.2022.4.11.

[33] Pethuru Raj et al. The Internet of Things and Big Data Analytics: Integrated Platforms and Industry Use Cases. CRC Press, 2020. **DOI**: 10.1201/9781003036739.

[34] U. Raza, P. Kulkarni, and M. Sooriyabandara. "Low Power Wide Area Networks: An Overview". In: *IEEE Communications Surveys Tutorials 19.2* (2017), pp. 855–873. **DOI**: 10.1109/COMST.2017.2652320.

[35] Sudhir K. Routray et al. "Satellite Based IoT for Mission Critical Applications". In: *2019 International Conference on Data Science and Communication (IconDSC)*. 2019, pp. 1–6. **DOI**: 10.1109/IconDSC.2019.8817030.

[36] Olivier Ruas. "Neighbor-of-Neighbor Routing In Small-World Networks With Power-Law Degree". PhD thesis. INRIA-IRISA Rennes Bretagne Atlantique, équipe ASAP, 2013.

[37] Rashmi Sahay, G. Geethakumari, and Barsha Mitra. "A novel Network Partitioning Attack against Routing Protocol in Internet of Things". In: *Ad Hoc Networks 121* (2021), p. 102583. ISSN: 1570-8705. **DOI**: 10.1016/j.adhoc.2021.102583.

[38] O. Salman et al. "An Architecture for the Internet of Things with Decentralized Data and Centralized Control". In: *2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA)*. 2015, pp. 1–8. **DOI**: 10.1109/AICCSA.2015.7507265.

[39] Zach Shelby and Carsten Bormann. *6LoWPAN: The Wireless Embedded Internet*. Vol. 43. John Wiley & Sons, 2011. **DOI**: 10.1002/9780470686218.

[40] G. Simon et al. "Dependable k-coverage Algorithms for Sensor Networks". In: *2007 IEEE Instrumentation Measurement Technology Conference IMTC 2007*. May 2007, pp. 1–6. **DOI**: 10.1109/IMTC.2007.379153.

[41] Rahim Taheri et al. "Similarity-based Android malware detection using Hamming distance of static binary features". In: *Future Generation Computer Systems 105* (2020), pp. 230–247. **DOI**: 10.1016/j.future.2019.11.034.

[42] G. Tangari et al. "Self-Adaptive Decentralized Monitoring in Software-Defined Networks". In: *IEEE Transactions on Network and Service Management 15.4* (2018), pp. 1277–1291. **DOI**: 10.1109/TNSM.2018.2874813.

[43] Jean-Philippe Vasseur. "Terms Used in Routing for Low-Power and Lossy Networks". In: *RFC* 7102 (2014), pp. 1–8. **DOI**: 10.17487/RFC7102. URL: https://doi.org/10.17487/RFC7102.

[44] Abhishek Verma and Virender Ranga. "Security of RPL Based 6LoWPAN Networks in the Internet of Things: A Review". In: *IEEE Sensors Journal 20.11* (2020), pp. 5666–5690. **DOI**: 10.1109/JSEN.2020.2973677.

**Basma Mostafa** received a dual Ph.D. degree in Computer Science and Operations Research in 2019 from the University of Montpellier, France, and the Faculty of Computers and Artificial Intelligence, Cairo University, Egypt. She received the B.S. and M.S. degrees in Operations Research from the Faculty of Computers and Artificial Intelligence, Cairo University, in 2008 and 2013, respectively, where she is currently an Assistant professor in the Department of Operations Research. Dr. Mostafa was awarded the 2017 "L'Oréal–UNESCO For Women in Science Levant and Egypt" for her research on developing optimized models for monitoring IoT networks. Her research activities focus mainly on combinatorial optimization, network optimization, linear and integer programming, modeling, and simulation.

**Miklos Molnar** received the graduation degree from the Faculty of Electrical Engineering, University BME, Hungary, in 1976, the Ph.D. degree in computer science from the University of Rennes 1, France, in 1992, and the French HDR degree in 2008. He has been with the University of Montpellier, France, since 2010. He is a Professor Emeritus with the Department of Computer Science in the laboratory LIRMM of Montpellier. His research activities are in combinatorial optimization, network design, and optimization algorithms and tools. He conducted several studies to find dependable routes for sensible communications, efficient multicast routes, energy-aware k-coverage, routing protocols, and different optimizations in ad hoc wireless networks. Dr. Molnar participates as a PC member in the organization of conferences and on the Editorial Board of several journals.

**Mohamed Saleh** received a master's degree from Bergen University, Norway, the M.B.A. degree from the Maastricht School of Management, The Netherlands, and the Ph.D. degree in system dynamics from the University of Bergen, Bergen. He is a Professor and the former Head with the Faculty of Computers and Information, Department of Operations Research and Decision Support, Cairo University, Egypt. He is also an Adjunct Professor with the System Dynamics Group at the University of Bergen. He has authored or co-authored numerous papers in several international journals and conferences. He is currently the Manager of the Virtual Center of Excellence for Data Mining and Computer Modeling, Cairo University. His current research interests include system dynamics, simulation, futures studies, and management science. Dr. Saleh was a recipient of the IBM Faculty Award.

**Abderrahim Benslimane** received the B.S. degree in computer science from the University of Nancy, France, in 1987, and the DEA (M.S.) and Ph.D. degrees in computer science from the Franche-Comte University, France, in 1989 and 1993, respectively. He has been a Full Professor of computer science with Avignon University, France, since 2001. He has recently been a Technical International Expert with the French Ministry of Foreign and European Affairs, from 2012 to 2016. Dr. Benslimane received the French Award of Scientific Excellency from 2011 to 2014. He is an Area Editor of Wiley Security and Privacy Journal and an Editorial Board member of IEEE Wireless Communication Magazine and Elsevier Ad Hoc. His current research interests include the development of secure communication protocols for vehicular networks and the Internet of Things.

**Sally Kassem** received her graduate degree in 1998 and an M.Sc. degree in industrial engineering from the Faculty of Engineering, Mechanical Design and Production Department, Cairo University, Cairo, Egypt, and a Ph.D. degree in industrial engineering from Concordia University, Montreal, Canada, in 2011. She has been an Assistant Professor with the Faculty of Computers and Information, Department of Operations Research and Decision Support, Cairo University, since 2012. She is also an Assistant Professor at the School of Engineering and Applied Science, Industrial Engineering Program, Nile University, Egypt. Her current research interests include mathematical modeling and optimization, linear and integer programming, operations research methodologies, supply chain, logistics, and modeling and simulation.

# Application-Aware Analysis of Network Neutrality: A Scalable Real-Time Method

Péter Orosz, Tamás Skopkó, Tamás Marosits

*Abstract*—**Internet access subscribers expect a satisfying quality of experience for any accessed service, independently from time, place, and service- and content-type. Besides the ever-increasing amount of Internet data, the spectrum of video service platforms offering sharing and streaming also got significantly more comprehensive. Internet access providers try to avoid the exhaustion of network bandwidth by investing in network capacity or setting up higher-level resource management within their infrastructure. The primary question in this domain is how resource management constrains the subscriber to access an arbitrary service and experience good service quality. This question directly relates to network neutrality fundamentals.**

**This paper presents a real-time full-reference objective method to assess network neutrality. It contributes three novelties to support user-centric analysis of potential restraints affecting Internet access quality: i) the proposal supports application-specific measurements and involves real content and real traffic, ii) the measured traffic originates from the content provider's cloud infrastructure, iii) reference is created in real time. Accordingly, the proposal introduces a novel measurement layout. The key component is the emulated client that provides the real-time reference by emulating the access properties of the real client and accessing the same content simultaneously.**

**We demonstrate the method's feasibility with an application-aware proof-of-concept use case: video streaming from a public VoD provider. We have validated the method against the emulated network parameters using an extensive series of laboratory measurements.**

*Index Terms*—**Network neutrality, quality measurement, video streaming, objective quality model.**

## I. INTRODUCTION

CLOUD-BASED services have become the dominators of global Internet communication in the last decade. Evolved data-center technologies and architectures opened the way toward centralized, global service platforms. From an Internet provider's perspective, it is a crucial challenge to manage network resources optimizing its subscribers' quality of experience for those cloud services that dominate their traffic mix. Its endeavor to control resources, i.e., traffic engineering based on service popularity, may offend the original best-effort paradigm of the global Internet. Handling a wide scale of traffic types requires tools and techniques that support the network to fulfill various transmission requirements. Since the best-effort communication model effectively fosters new services and technologies, prioritizing the service platforms based on popularity ranking may set up numerous constraints to new technologies and services. The primary question in this domain is how advanced resource management constraints the subscriber to access an arbitrary service and experience good service quality. This question directly relates to network neutrality fundamentals which are supported by the legal framework Regulation (EU) 2015/2120 of the European Parliament and the Council [1]. The regulation and the status quo imply a pivotal question: How can consumers and authorities objectively verify Internet access neutrality for a wide range of services in the cloud-era?

Although the regulation itself is intended to be the legal mechanism in this field, a complete consumer protection suit also requires a dedicated technological background to enable verifying the neutrality of an Internet access, and the latter is still missing. The primary reason for this large gap between the availability of legal and technical tools is that the scientific fundamentals providing the underlying measurement paradigm and scalable methods are yet to be established. Analyzing a wide scale of network neutrality attributes (defined by EU's BEREC Office [2] [3] and introduces new ones for cloud services) is a research "green-field" without dedicated standards and methods.

The network neutrality paradigm expects Internet access providers to treat all user traffic equally, independently from the type of device, platform, service, and content [1]. In contrast, the last decade has seen many user restrictions and traffic differentiation cases that violate the neutrality principle. Early cases typically covered service functionality restrictions by explicitly blocking domains, IP address ranges, and transport protocol port identifiers. With the evolution of deep packet inspection (DPI) technologies, operators can identify a service while its traffic enters their networks. The service identification, whether it is based on protocol identifiers or traffic pattern recognition, enables performing prioritization. As a result, some preferred applications may offer guaranteed perceptive quality by assuring bandwidth or latency. Other services use the rest of the network resources, thus being transmitted according to the best-effort model or even with administratively limited throughput.

For assessing the service (and mainly audio and video) quality, objective quality models can be categorized into two major categories, i.e., full-reference and no-reference models. While the full-reference models require a reference source to perform quality analysis on the received media and typically have a high correlation to the perceived service quality, the no-reference models do not use such references but have lower accuracy. Applying full-reference models to assessing public services is problematic. The central question is how a valid

P. Orosz, T. Skopkó and T. Marosits are with the Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics (BME), Budapest, Hungary (e-mail: orosz@tmit.bme.hu, skopko@tmit.bme.hu, marosits@tmit.bme.hu).

Manuscript received September 5, 2022;

DOI: 10.36244/ICJ.2023.1.8

reference source can be obtained. The primary disadvantages of the full-reference model are a) offline operation (i.e., post-processing), b) requirement for the original sample (as a reference) and as a fundamental privacy issue, c) analysis of user data. Based on the new methodology, we propose a real-time full-reference assessment model to identify service quality restrictions the Internet access provider applies.

Considering the assessment of network neutrality (and particularly with an application-specific focus), objective assessment methods can be categorized into three major categories based on the applied traffic pattern, i.e., real, replayed, and generated traffic. The two central questions are: i) How an assessment method relates to the real user scenarios and traffic patterns? ii) How can false detection be identified, and what is the probability of a false result? Network neutrality measurements on the public Internet cannot be considered a repeatable, all-the-way objective measurement from the metrological perspective. A comprehensive neutrality analysis covers a wide scale of measurement types that should be performed reliably on a live network. While the protocol-based test can be executed with a relatively low false ratio, application-specific measurements raise multiple methodology-related issues.

The primary goal of this paper is to establish a new assessment paradigm - real-time application- and platform-specific analysis of network neutrality on the public Internet (assessment anytime, anywhere). The involvement of public cloud services enables the elaborated objective models to continuously adapt to the ever-changing public networks and platforms by exploiting the benefits of real service traffic. Moreover, the novel measurement paradigm allows the assessment of network neutrality for a wide range of applications with a low false rate.

The remainder of the paper is organized as follows. Section II presents the related works in the field of technology- and application-specific analysis of network neutrality. Section III introduces the new methodology focusing on the novel principles, and Section IV describes the assessment method that aligns with the criteria specified in BEREC measurement directives [2] [3]. In contrast, Section V presents a proof-of-concept use case applying the method to detect video streaming restrictions. Section VI shows validation results to demonstrate the effectiveness of our model. Finally, Section VII concludes the paper.

## II. Related Work

Previous works on network neutrality assessment, in general, try to detect possible differentiation with a manipulated user traffic pattern and compare the metrics of the transmissions [4] [5], being the latter the reference. Some of them aim to detect the shaper algorithm and its parameters as well [6] [7]. The real challenge for all of them is to lower the false positive ratio (differentiation detected, but there is none) and false negative (differentiation stays under the radar) detection.

NetPolice's idea uses two flows between the endpoints: the original is the reference, and a generated one with a similar timing but with different ports and payload is to be replayed [8]. The flows are transmitted among similar packet loss if no shaping is applied between the endpoints. Routing information is also considered. Since it is based on ICMP replies of the hops, its usability is affected by rate-limiting and protocol prioritization.

Glasnost uses real traffic and its randomized copies with similar timing [9]. To detect shaping, these copies are replayed via the measurement servers at the same or higher rates than the original one. Maximum application throughput is compared with application control flow throughput as well. To improve its accuracy, user traffic traces were collected and analyzed; different flow types were compared on the same network path, as well as measurements were executed at least five times and at least for 60 seconds per use case. It used a dedicated application (discontinued).

DiffProbe also uses packet replay at different rates, and link saturation [10]. It takes packet loss and end-to-end delay into account when detecting possible differentiation. Its limitation is that it also classifies by port numbers and payload type since behavior-based differentiation methods were unreliable at the time.

OONI's goal is detecting Internet surveillance and censorship [11]. Using active probing, it tries to connect various HTTP-based services. It also utilizes DNS lookup and multi-protocol traceroute, detects TCP resets, and man-in-the-middle (MITM) SSL/TLS interventions. However, more of these tests affect the application level; it does not perform application or service-specific tests.

Network tomography and inference framework use a different approach [12]. It measures TCP and UDP traffic flow congestion between a set of endpoints and constructs a linear system of equations. An unsolvable system means positive detection. However, this solution needs extensive infrastructure and a large number of users. The advantage is a low false detection rate and keeping count of TCP dynamics.

Researchers behind CONNEcT realized that identifying network neutrality measurements can help ISPs divert the results [13]. The detection is based on packet loss and passive path capacity measurement. Their solution uses a covert communication channel between measurement hosts: these metadata and samples are hidden within the application data. No application-specific measures are made, however.

WindRider focuses on measurements between its measurement servers and mobile endpoints [14]. Although measurement is lightweight, the method requires user feedback, and demands are special on OS features (packet capturing, etc.) that limit the wide implementation. The user feedback does not judge an application's QoE (Quality of Experience) itself.

Statistical analysis is commonly used to minimize the effect of background traffic as well as to reduce the false detection ratio with more or less success [15]. Only a few methods use real application traffic during the measurement, and none of them use QoE as the base of their detection. Some of the methods above require endpoints placed at other ISPs or dedicated applications or frameworks on the endpoints. If a measurement server is used, ISPs can also recognize and whitelist the measurement traffic.

## III. METHODOLOGY

### A. Measurement on live networks

Our measurement method approaches the detection by the real user QoE. First, the user's access parameters are determined. Then the measurement is executed with two instances simultaneously: it is measured at the user endpoint and in a container with the same access parameters at an Internet exchange point (IX). Since IX can be considered a neutral access point, the measurement results will be used as a real-time generated reference for the measurement. The measurement itself measures the specific service in subject instead of only replaying a recorded traffic pattern and measuring QoS metrics. Despite the ISP being inspected could detect the measurement control server's IP addresses, the baseline of the measurement cannot be altered.

Measurements performed on a live public network (i.e., the Internet) are not repeatable in the sense that we execute measurements in a controlled laboratory environment. Accordingly, evaluating the measurement results requires a new perspective and alternative methods. Whether the assessment involves packet-level or application-level performance metrics, consecutive measurements never give the same result, even between two designated endpoints. In order to improve the reliability of the outcome, there are existing methods to filter out traffic interference and other endpoint-related constraints [16] [17]. On the one hand, there exist measurement methods supporting a single measurement by eliminating the effect of transient events that occurred during the assessment process. Finally, there are statistical methods to handle anomalies based on a large number of measurements. When such an assessment tool is made publicly available, we must handle the most challenging scenario, i.e., the network neutrality assessment is allowed to perform by anybody using his/her computer and Internet access. The primary question this scenario raises is how we can get a reliable result, possibly free from false detection or identifying an invalid case at least. From the metrological perspective, the main task is to minimize measurement error even with these constraints. Accordingly, we will discuss the factors that affect measurement reliability and may induce false detection on a public network.

### B. Handling false detection

Focusing on the root cause of false detection, we can categorize the sources of unwanted events and properties into two categories: i) traffic interference and ii) resource interference. While the former covers background traffic that is concurrently transmitted on any segment of the IP path during the measurement process, the latter refers to the host's hardware resources (e.g., CPU and system memory) that are shared between running processes, including the measurement itself. While lightweight background traffic and concurrent processes do not affect the reliability of the measurement result, heavy workloads may induce false detection of traffic manipulation. Verifying resource availability is a pivotal step to minimize the probability of false detection. We propose a preliminary measurement phase for a web-based approach that aims to estimate the available access bandwidth and the packet loss ratio. Alternatively, an application-based approach also enables to verify CPU, memory, and bandwidth availability preceding the network neutrality measurement.

### C. Causes of detection uncertainties

*1) Background user traffic:* Background traffic may alter the time-domain behavior of the measurement traffic and thus affect the measured transmission properties. Since the presence of background traffic is inevitable in public network measurements, our goal is to minimize its effect on the measurement result. In the worst-case scenario, excessive background traffic congests a network link, resulting in packet loss in the measured traffic. These loss events directly affect the application-level performance of the measured service.

*2) Process-level interference on client-side resources:* Available CPU cores, system memory, and Internet access capacity are shared resources between concurrently running system and user processes.

*3) Congestion on the content provider's infrastructure:* Our proposal involves the content provider's infrastructure in the measurement process. Accordingly, bottlenecks occurring on the provider-side directly impact the measurement outcome. However, cloud-based virtualized content services scale well by design to maintain high service quality even in peak time periods. Meanwhile, specific scenarios may occur when service quality drops due to unexpected security or resource provisioning issues. Our proposed dual test method can identify quality degradation originating from a bottleneck that evolved in the provider's infrastructure.

### D. Whitelisting the measurement traffic

Most of the discussed proposals are based on the client-server communication model. The drawback of this architecture is the static nature of the server-side, which is easily identifiable via its IP address. This enables Internet Service Providers to dynamically detect and prioritize measurement traffic at packet-level in real time.

Solution: Since the proposed method fetches real content from the content provider, the server-side measurement IP address belongs to the provider's IP network. This feature disallows ISPs to identify the measurement traffic on their networks. However, a single TCP connection permanently exists between the measurement client and the control server throughout the entire measurement session. While the server's IP address is fixed, it can be masqueraded by deploying one or multiple relay servers (exclusively for the control messaging) in an arbitrary public cloud. In this case, ISP can only identify the IP of the relay server that belongs to the public cloud provider.

## IV. THE PROPOSED METHOD

### A. Feature overview

Our proposal has three key novelties: i) it is horizontally scalable to any cloud-based service, ii) it measures real traffic originating from a public service provider, and iii) measurement reference is created concurrently in real time. While

we proved the feasibility of the proposal using cloud-based services, there are no technological constraints limiting its usage to web-based applications on the client side. The primary goal of the method is to perform application-specific network neutrality measurements on the public Internet reliably.

### *B. Measurement architecture*

The architecture incorporates three major components to support the dual measurement concept: the real client, the measurement server, and the content provider service (see Fig. 1).

Every neutrality measurement is performed in parallel: the real client and its emulated twin (imitating the Internet access parameters of the real one) simultaneously measure the same service. The real client is using the neutrality measurement application on its host, and the emulated client is running the same application in the emulated container on the server.

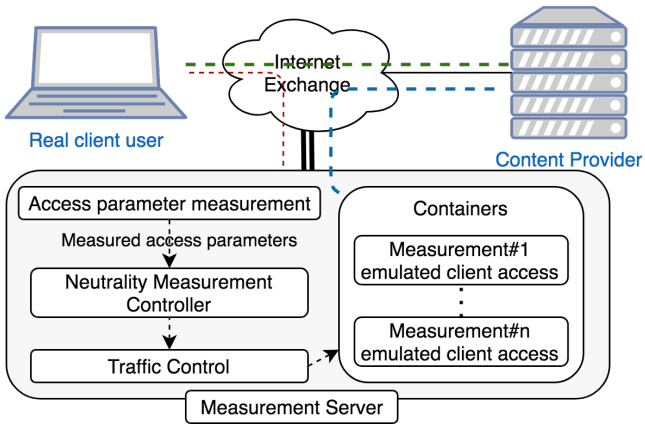



Fig. 1: The proposed measurement layout and architecture

The measurement server is a generic server architecture for running the software components required for the measurement process: performing Internet access parameter measurements for the real clients, creating network containers for Internet access emulation, executing service measurements for the emulated clients, as well as controlling the real client during the process. The server should have an uplink (marked with a double black line) broad enough for measuring all possible ISP access profile types (e.g., at least 10 Gbps if we want to measure multiple clients with 1 or 2 Gbps access). After bandwidth measurement, real clients exchange only control messages with the measurement server. This logical connection is marked with a red dashed line. The service data stream between the real client and the content provider is marked with green, and between emulated clients and the provider is marked with blue dash lines.

### *C. Real-time reference: dual measurement with client emulation*

The most crucial component of the measurement architecture is the emulated client that adopts the principal network access properties of the real measurement client: download and upload bandwidth, round-trip time (RTT) latency, and optionally packet loss ratio. Furthermore, the measurement server should be deployed to a location independent of the user's Internet Service Provider, i.e., to the logical proximity of an Internet Exchange Point. A preliminary QoS measurement phase can determine the network access properties of the real client.

### *D. Measurement workflow*

The Neutrality Measurement Controller (see Fig. 1) manages the measurement by instructing both the real and the emulated clients. The following phases are executed during a measurement session:

1) Preliminary QoS measurement on the real client to determine the major network access parameters.
2) The emulated client adopts the access profile of the real client by software-based emulation of the same QoS parameters within a network container.
3) Both clients initiate the measurement by requesting the same content from the content provider.
4) While the content is streamed online, both clients measure the key performance indicators. We note that the set of appropriate performance indicators should be uniquely defined for each service (see Section V for proof-of-concept use cases).
5) Both clients independently evaluate service quality based on the measured KPIs.
6) Processing the evaluation results, the measurement controller determines the overall service quality and represents it to the user on a predefined quality scale (e.g., the ITU P.800 MOS scale [18]).

The preliminary QoS measurement incorporates a method developed in one of our previous research and development projects (SCL Broadband Measurement System [1]). This system is validated and applied by the Hungarian National Media and Infocommunications Authority as a reference for measuring Internet access parameters. See Section VI for validation details.

Handling false detection: Including a real-time measurement reference, our method also enables identifying bottlenecks on the content provider-side.

## V. PROOF-OF-CONCEPT USE CASES

This section introduces a video-on-demand use case for Youtube to prove that our method is feasible and applicable to a wide range of cases. Though, the neutrality measurement should be uniquely implemented for each service or platform. A measurement session starts with a quick estimation of user access parameters: upload and download bandwidth is determined, as well as packet loss and RTT. These parameters will be used to create a measurement container on a measurement host located at a neutral Internet access point. Reference measurements are then executed within this container, and it will be called emulated client. As mentioned previously, we

[1] https://szelessav.net/en/

examine the service's quality at the real and the emulated client in parallel.

We had to look for the measurable and gradable phenomenon of the use case to correlate them with QoE. However, we should only consider phenomena whose cause is the QoS degradation of the user's Internet access or degradation of the inspected service itself.

Examples of such a phenomenon: playout buffer becomes empty, video/audio stream quality changes (degrades), video/audio stream source bitrate changes (decreasing bitrate during adaptation), and video/audio decoding errors during the playback.

As we measure services in the browser at the user endpoint, we had to rely on the streaming service's API (Youtube, in this case). We chose the iFrame Player API [19] because it provides functions for measuring the phenomena mentioned above.

The following functions and events can be considered helpers for the measurements:

- player.getVideoLoadedFraction():Float – queries the downloaded ratio of the entire stream.
- player.getCurrentTime():Number – queries the seconds elapsed since the start of the playback.
- onPlaybackQualityChange – the event triggered when (the class of) playback quality changes.

Accordingly, we can construct a metric based on the ratio of the time when the playback buffer is empty against the entire playback time. Alternatively, the metric can express the time ratio of the degraded playback quality against an explicitly requested media quality (e.g., playing back in SD instead of the requested 720p). Then, we can apply a weighted linear combination of them.

We also had to consider the media playback process in the user's browser. Each playback starts with a pre-fetch phase when the application loads the first part of the video stream to the playout buffer, then it switches to a playback phase. Supposing that the user equipment does not have resource constraints, the playback only gets stalled when the playout buffer becomes empty. We can query the downloaded ratio of the stream by the *player.getVideoLoadedFraction()* function at any time. This function reports the downloaded proportion independent of the video quality and size. Since we know the video's duration, we can transform the loaded fraction value into a time value to express the time position until the video was already downloaded. We will call this *video time*. Of course, parts of this downloaded content are already decoded and rendered, but another part of the undecoded bitstream may reside in the playout buffer. The *player.getCurrentTime()* function reports the time elapsed since the playback started. If we look at the time function of these two metrics, a continuous playback's time can be depicted by a line, but the loaded amount (video time) is a monotonic growing curve, as seen in Fig. 2.

We marked the buffer status in green and the playback time in red. The buffer becomes empty when playback time is not strictly monotonically increasing or rising above the video time. In the presented scenario the green curve is always above the red; that is buffer never gets empty.

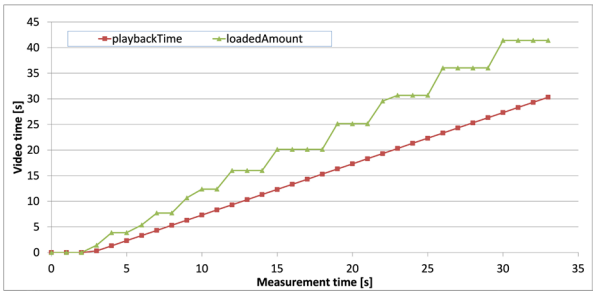



Fig. 2: Relation between playback time and video time on a 4/1 Mbps connection

By continuously sampling the playback process, the Youtube playback QoE can be estimated by the number of samples with continuous playback (playout buffer never gets empty) against the total number of samples, as of (1). Here, the playback QoE indicates network neutrality, i.e., Network Neutrality Index (NNI).

$$NNI^{1}_{str} = \frac{\text{\# of samples of continuous playback}}{\text{total \# of samples}} \quad (1)$$

Besides the measurements executed locally in the user's web browser, we can indirectly rely on the Youtube service's download controller reports. These reports aim to inform the content provider about the users' communication environment and help optimize the download process to improve the QoE. One feature of this optimization is the automatic selection of the bitstream format. The download controller is notified via the *onPlaybackQualityChange* API event. The event reports a quality class string, but indirectly refers to the video bitstream format. The class identifies a subset of bitstream formats, e.g., hd720p includes bitstream formats encoded with various profiles of avc1 and vp9, all being 720p resolution. Thus, we cannot determine the current bitstream format, bitrate, framerate, etc., or the exact source data rate. Still, we have received an explicit notification about the client's Youtube download controller; what is the best achievable quality class that the client can receive besides the current (previously measured) access parameters. A change in quality can denote a change in available bandwidth and thus can be considered a service quality metric. This event hook also completes our estimation method without being limited to periodic sampling.

Fig. 3 shows the relationship between the abovementioned status information. The graph also shows that decoding the video content (i.e., rendering) can cause a performance bottleneck at the user endpoint, limiting the achievable quality, even though the quality of raw data transmission is appropriate. This test was executed at the bandwidth of 100 Mbps in both directions, while the 8K video content's source data rate does not exceed 25 Mbps. Thus, available access link bandwidth was not a limiting factor. We also repeatedly experienced stalled downloads and noticed that the download controller switched to a lower-quality class. For example, on Fig. 3 at 33 seconds, the playback stopped (see red graph), then the buffer was purged, and after that, from 38 seconds, the buffer

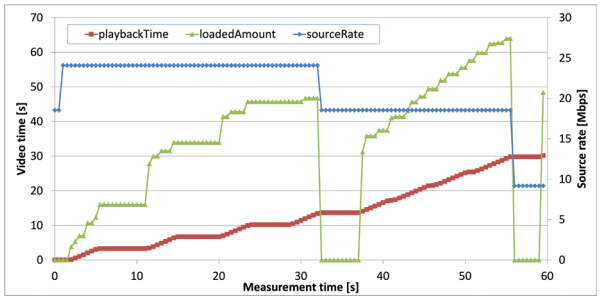



Fig. 3: Symptoms of video quality class change in function of time on a 100/100 Mbps connection

size started to increase again (see green graph) but the source rate - which relates to the video quality - was lower than before (see blue graph).

The *onPlaybackQualityChange* event occurrs whenever the quality class is changed during the playback, allowing us to perceive the change immediately. This concludes with a more precise reconstruction of the changes during the playback. We can derive a metric from these series of events providing the basis for the network neutrality index of the Youtube video streaming service. A simple case of the metric relating to the violation of network neutrality is presented by (2). The target format is the same or better class than the auto-selected one during the playback start. Accordingly, the metric is calculated by the number of good-quality samples (i.e., that are not scaled down) against the total number of samples.

$$NNI^2_{str} = \frac{\text{\# of samples in target format}}{\text{total \# of samples}} \tag{2}$$

We can construct measurement results from any of the metrics discussed previously. For a more precise estimation, we decided to use the sum of equally-weighted normalized values as of (3).

$$NNI_{str} = 0.5 \times NNI^1_{str} + 0.5 \times NNI^2_{str} \tag{3}$$

Of course, the same formula must be used for both the real and the emulated client. Suppose the reference (emulated client) value is lower than the real client's one. In that case, we can suspect that degradation is not caused by the user's Internet access but a bottleneck at the content provider. To create an index value that is easier for humans to interpret, the NNI value can be formed between 1 and 10 as of (4).

$$NNI^{human}_{str} = 10 \times (0.1 + 0.9 \times NNI_{str}) \tag{4}$$

## VI. Validation

The proposed method relies on measuring the network access properties of the client and applying it to the emulated client to concurrently access content available at the same content/service provider. Accordingly, the performance of the QoS emulation directly impacts the validity of the measurement reference provided by the emulated client. Since the network emulation incorporated download and upload bandwidth and latency, the primary aim of the validation is to compare the traffic properties of the real and emulated clients. This section presents the validation results of the QoS emulation. We can measure the effectiveness of the emulation by analyzing the similarity between the two traffic patterns. In our proof-of-concept system, the emulation subsystem is based on the Linux kernel-based Network Emulator (NetEm).

Three criteria of the validity: 1. Independent Internet access (Internet Exchange), 2. Valid QoS emulation of the access properties, 3. Valid measurement reference.

Our method uses a trusted reference measurement executed parallel with the user's measurement during network neutrality-related service inspection. The measurement hosts use container technology to emulate the user's access parameters properly. Our method can be considered valid if the emulation is working correctly. Access parameters are measured on a link without background traffic. Bandwidth, bi-directional latency (RTT), jitter, and packet loss are measured.

Our proposal introduces a novel application-specific measurement method incorporating a unique approach for the measurement architecture as well. In contrast, previous works on network neutrality assessment, in general, try to detect possible differentiation with a manipulated or generated user traffic pattern. Therefore, they cannot be effectively compared to our proposed solution.

### A. Measurement setup

For validation, a setup with two hosts connected directly was used. One host is a bandwidth measurement system, and the other one runs the browser in a container with specific access parameters. Both hosts are capable of generating traffic of at least 1 Gbps.

Fig. 1 details the measurement system's structure. The dedicated container is created each time a measurement has to be run. At the beginning of the user's measurement, a quick quality of service assessment is run on the client side. This phase determines its network access parameters (download and upload bandwidth, delay, and packet loss rate). These access parameters are used to create a similar environment for the neutrality measurement at the initialization of the container.

The container is based on Linux Network Namespaces (netns). It is a Linux kernel technology builder for software containerization. A netns container has its own network interface but can be run normally with the same computing resources anyway. A point-to-point network connection is established between the system and the container.

Linux Traffic Control (tc) is a generic tool used to configure shaping and policing but also supports scheduling and dropping. The proper configuration using the *tc* command can emulate a link with specific access parameters. Since the point-to-point connection is built between two virtual network devices, each direction can be configured independently; thus it can also emulate an asymmetric connection.

For the more sophisticated realization of specific network access, we applied Linux NetEm (netem) scheduler since it allows us to add delay, jitter, and packet loss.

After the container is set up, the neutrality measurement is run simultaneously on the real client and the reference

container. If the network access is emulated correctly and the client has neutral Internet access, similar results should be obtained from both parties.

### B. Validation methodology

During the measurement sessions, we iterate through a set of access parameters. After a session, the measured results are compared to the nominal access parameters: average and deviation are calculated to parameters bandwidth, delay, and packet loss. We expect the values to be within an error margin of 4%.

RTT is measured by generating consecutive websocket ping-pong messages. These packets are sent within TCP packets with an urgent flag and are replied to immediately by the receiver's websocket stack. Minimum and maximum RTTs are determined, as well as the average and the standard deviation of delays.

Throughput is measured by saturating the link in the affected direction by using the appropriate number of parallel websockets. The transfer is started with only one socket to avoid overloading narrow links. If per-socket throughput is over a target rate, new sockets are opened to try to saturate the link.

Packet loss is measured using TCP analysis: TCP segments are traced, and if a segment or its part is re-transmitted, it is calculated as lost. We derive packet loss from the amount of re-transmitted data against the total bytes transferred.

### C. Container validation

The quality of service parameters of the connection seen by the container was validated with the Broadband Measurement System of the National Media and Infocommunications Authority[2]. Thus, we proved that the connection available for a container created in a net-neutral environment is equivalent in terms of quality of service to the connection that the user of the net-neutral metering system receives from its service provider. The accuracy of this Broadband Measurement System and the stability of the measurement series under laboratory conditions were validated by measurements at known native interface speeds and by using the Spirent Attero X network emulator as a bandwidth and delay reference. According to the measurements, the error of the internet speed measurements is under 2% between 100 kbps and 2 Gbps if the round-trip connection delay does not exceed 100 ms or the BDP (Bandwidth Delay Product) does not exceed 4 MB.

*1) The correct value for the object to be measured:* In the tests, we use maximum-sized Ethernet frames without 802.1Q VLAN tags to measure throughput, so their length is 1538 bytes in the physical layer. Because each packet contains a 12-byte TCP option after the TCP header, the maximum length of the TCP segment embedded in the IP packet within the Ethernet frame is only 1448 bytes. Overall, the throughput can be at most 1448/1538 = 94.1482% of the physical bit rate at some native interface speed if the IFG (Inter-Frame Gap) is not reduced. Suppose the speed limit is set with NetEm, and this limit is interpreted in the data link layer. In that case, the theoretical maximum is 1448/1514 = 95.6407% of the set value if we do not consider the possible reduction of the IFG. The presented measurements were performed uniformly over HTTPS.

Regarding the delay time, it should be noted that when creating a packet delay on the prepared virtual interface, NetEm even adds to the set value the time it takes for a packet of the same size to pass through an interface with the same physical speed. On HTTPS, we use 119-byte Ethernet frames from either the client or server for delay measurement. This would result in another 24 bytes in the physical layer (IFG + Preamble + Start-of-Frame-Delimiter + CRC), but this should not be considered, as NetEm's rate limiter settings are interpreted in the data link layer. Thus, the combined packet forwarding delay of packets passing through the interface in one direction or another, for example, on a 1 Mbps symmetrical connection 1.904 ms, which corresponds quite well to the excess experienced, so we can correct the value of the measured delay with it. In the higher speed range, the effect is no longer significant.

*2) Measurement results:* In addition to the mean ($m$) and standard deviation ($\sigma$) of the bidirectional delay, our data include the mean and median of the cleaned set of samples per second of download and upload speeds in the transport layer, as well as the packet loss rate detected in each direction. For us here, the mean and standard deviation of the bidirectional delay and the downstream and upstream throughput will be the most interesting. The following is a table-form presentation of some of the measurement data, namely the mean and standard deviation of the measurement results. At least 50 measurements were made at each setting.

The presented results in Tables I-III prove that we can provide a network connection to the reference container in a predictable way according to the interface emulation settings. Table I provides the statistics of the measurements of a container with 1 Mbps symmetric bandwidth, Table II presents similar statistics of a container with 100 Mbps symmetric bandwidth. In contrast, Table III provides the statistics of 1 Gbps symmetric bandwidth container. These bandwidths were set in the NetEm. The first column of the tables shows the intended bidirectional delay, also set in the emulator.

Table headers shorten the followings:

- $D^{set}$ is the delay in *ms*, which was set as the interface delay of the container's virtual interface. This delay was symmetrically distributed.
- $RTT^{meas}_{avg}$ is the average of measured end-to-end delay samples on the container's virtual interface in *ms*.
- $RTT^{meas}_{stdev}$ is the standard deviation of measured end-to-end delay samples on the container's virtual interface in *ms*. This can be considered as the delay jitter.
- $DSrate_{avg}$ is the average downstream rate on the container's virtual interface in *Mbps*.
- $USrate_{avg}$ is the average upstream rate on the container's virtual interface in *Mbps*.

While $DSrate_{avg}$ and $USrate_{avg}$ are measured in the transport layer, they represent the TCP throughput the user in each direction can achieve.

[2]https://szelessav.net/en/

The cells of Tables I-III contain two numbers. As it is mentioned in the 2nd column of the tables, the upper number is the average of the measured quantity identified by the header of the actual column. In contrast, the lower number is the standard deviation of the same quantity. The averages in columns 3, 5 and 6 can be compared to the values preset in the emulator considering the argumentation mentioned above about the correct value for the measured object.

TABLE I
STATISTICS OF THE MEASUREMENT RESULTS AT 1 MBPS

| $D^{set}$ | | $RTT^{meas}_{avg}$ | $RTT^{meas}_{stdev}$ | $DSrate_{avg}$ | $USrate_{avg}$ |
|---|---|---|---|---|---|
| **0** | $m$ | 2.277667 | 0.089067 | 0.961765 | 0.952292 |
| | $\sigma$ | 0.029331 | 0.053768 | 0.00616 | 0.00168 |
| **1** | $m$ | 3.271317 | 0.097783 | 0.960243 | 0.952468 |
| | $\sigma$ | 0.038381 | 0.042514 | 0.007081 | 0.001556 |
| **4** | $m$ | 6.279983 | 0.085167 | 0.958726 | 0.952579 |
| | $\sigma$ | 0.029907 | 0.187768 | 0.006517 | 0.001949 |
| **10** | $m$ | 12.2851 | 0.08595 | 0.958213 | 0.95258 |
| | $\sigma$ | 0.024229 | 0.057395 | 0.007618 | 0.001532 |
| **30** | $m$ | 32.28538 | 0.1236 | 0.958262 | 0.952457 |
| | $\sigma$ | 0.032863 | 0.217209 | 0.007157 | 0.001705 |
| **100** | $m$ | 102.3016 | 0.089153 | 0.958991 | 0.953304 |
| | $\sigma$ | 0.015433 | 0.055426 | 0.00711 | 0.00142 |

TABLE II
STATISTICS OF THE MEASUREMENT RESULTS AT 100 MBPS

| $D^{set}$ | | $RTT^{meas}_{avg}$ | $RTT^{meas}_{stdev}$ | $DSrate_{avg}$ | $USrate_{avg}$ |
|---|---|---|---|---|---|
| **0** | $m$ | 0.205198 | 0.034887 | 95.65311 | 95.56265 |
| | $\sigma$ | 0.011178 | 0.004756 | 0.037675 | 0.012557 |
| **2** | $m$ | 2.388895 | 0.039735 | 95.65777 | 95.55468 |
| | $\sigma$ | 0.026957 | 0.061805 | 0.048672 | 0.014351 |
| **4** | $m$ | 4.394665 | 0.054578 | 95.64517 | 95.55992 |
| | $\sigma$ | 0.033135 | 0.106249 | 0.019685 | 0.014976 |
| **16** | $m$ | 16.39481 | 0.042604 | 95.64445 | 95.54124 |
| | $\sigma$ | 0.022767 | 0.032298 | 0.020738 | 0.01707 |
| **30** | $m$ | 30.39306 | 0.046067 | 95.63466 | 95.52246 |
| | $\sigma$ | 0.019898 | 0.026699 | 0.018684 | 0.027681 |

TABLE III
STATISTICS OF THE MEASUREMENT RESULTS AT 1 GBPS

| $D^{set}$ | | $RTT^{meas}_{avg}$ | $RTT^{meas}_{stdev}$ | $DSrate_{avg}$ | $USrate_{avg}$ |
|---|---|---|---|---|---|
| **0** | $m$ | 0.087567 | 0.023533 | 956.8405 | 956.4033 |
| | $\sigma$ | 0.012218 | 0.007524 | 0.784973 | 0.225397 |
| **1** | $m$ | 1.151567 | 0.025517 | 956.7349 | 956.31 |
| | $\sigma$ | 0.026311 | 0.009899 | 0.617618 | 0.19924 |
| **2** | $m$ | 2.19315 | 0.0341 | 956.7329 | 956.3524 |
| | $\sigma$ | 0.028002 | 0.03899 | 0.646558 | 0.18775 |
| **4** | $m$ | 4.197133 | 0.042383 | 956.7777 | 956.4288 |
| | $\sigma$ | 0.022469 | 0.055571 | 0.810262 | 0.284104 |
| **10** | $m$ | 10.19637 | 0.03395 | 956.8733 | 956.8491 |
| | $\sigma$ | 0.021568 | 0.013686 | 0.65082 | 1.374928 |
| **16** | $m$ | 16.19535 | 0.0404 | 956.7959 | 956.3924 |
| | $\sigma$ | 0.014769 | 0.046178 | 0.713448 | 0.26032 |
| **20** | $m$ | 20.1927 | 0.037117 | 956.7736 | 956.2828 |
| | $\sigma$ | 0.017342 | 0.014367 | 0.848646 | 0.169963 |

We can see that the rates are very close to their theoretical limits, which shows, that the emulation is very accurate. This statement is confirmed by the values of sigmas ($\sigma$) which are under 1% of the appropriate averages ($m$) at 1 Mbps and under 0.1% at 100 Mbps and 1 Gbps. The difference between the measured and the expected RTT - note that this later differs from the delay set in the emulator - is no more than 2%. Obviously, except for the delay set to 0, where the standard deviation of the average RTT can be higher. Column 4 presents the average and the standard deviation of the delay jitter series in the tested scenarios. Although NetEm would allow it, we do not use this as a configuration parameter of the container, but we measure it. As we can see, the average delay jitter is less than 1 ms in every scenario, which is considerable for real-time multimedia services and these values are stable.

As a summary of the container validation, it can be established that for both the statistics presented here and those omitted due to lack of space support that, we can infer the measurement result from the given settings with high reliability, i.e., the network emulation is accurate. The standard deviations show that the results are very stable.

### *D. Validation of the VoD (Youtube) assessment method*

Two different measuring devices - in this case, the arbitrary browser used as a client and the Chromium browser engine running in the node.js environment in the container - can be considered the same if the distribution of their measurement results is identical for a measured quantity. Instead of matching the distribution, we can also accept the sameness of the statistical indicators, provided that sufficient measurements are made.

Among the features presented earlier, the download rate returned by the *player.getVideoLoadedFraction()* function is what gives a monotonically increasing curve over time and can be considered a linear function of time. This offers the option of fitting a line to this with the least-squares error, as Fig. 4 shows. In the case of measurement, the fitted line can be given by its intercept and slope, while the measuring device is characterized by the statistical properties of the slope and intercept of the regression lines of the measurements in the measurement series. Based on the average of the axis intersections and the average of the slopes, we can draw a line - this will be referred to later as the "center line" - which can actually be interpreted as a measure of the current client configuration. However, we can derive the uncertainty of this measurement based on the standard deviations of the axial intersections and slopes. If we add or subtract twice the standard deviation of the mean slope and add or subtract twice the standard deviation from the average slope, we get two more lines, later referred to as "bounding lines", which delimit it on both sides. The part of the plane to which the fitted line of the individual measurements falls with a probability of 95%. In fact, we gave the characteristic of the current configuration of the measuring instrument, as shown in Fig. 5, where the center line is drawn in blue, and the upper and lower boundary lines are drawn in red, and green, respectively.

*1) Dependence of the reference client's parameters on the operating system:* Since Youtube traffic is over TCP[3]. There is a measurable difference between TCP implementations on different operating systems (e.g., Ubuntu Linux and Windows 10). So it is expected that the measurement results of the video streaming service testing module of the net-neutrality measurement system will also be affected by the client's

[3]Youtube also supports QUIC/UDP, but the container limitations related to Layer 2 traffic, thus they can be considered with the same impact in the case of UDP. The API used for the Youtube-measuring method mentioned above is independent of the transport protocol.

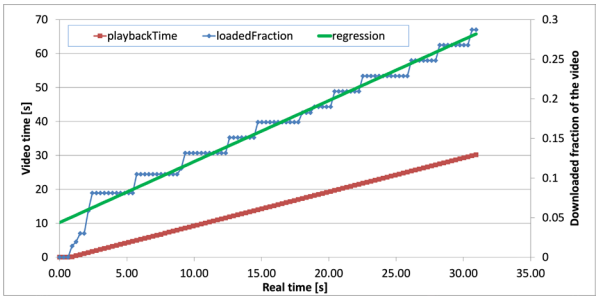



Fig. 4: Download rate curve with the line fitted to it (50/20 Mbps, Firefox/Ubuntu)

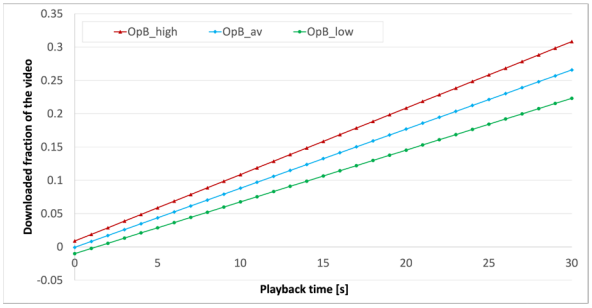
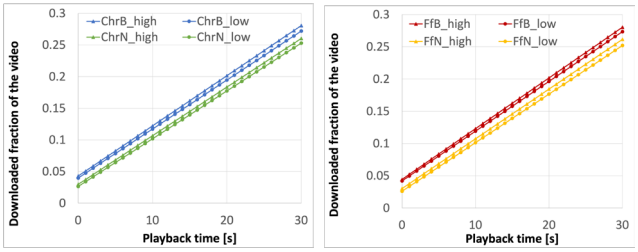



Fig. 5: Characteristics of the 100/100 Mbps connection (Opera/Ubuntu)

operating system. This permanent discrepancy is unavoidable because node.js runs the Chromium browser engine in the container replicating the client. However, based on our studies, this deviation is linear. Based on the information available in the measurement system, the permanent error can be reliably estimated, and then the measurement result can be corrected. In our experience, the extent of this discrepancy also depends on which browser, possibly which version, is used and the bandwidth of the connection available to the user.

*2) Dependence of the reference client's parameters on the browser:* In Fig. 6 below, we can see that when testing on Ubuntu Linux over a 100 Mbps/100 Mbps connection, both the Chrome browser and the Firefox browser outperform the container. However, it is also noticeable that the behavior of browsers is very similar to each other. Also, on Ubuntu, for example, looking at a 30 Mbps/10 Mbps connection, we see that the characteristics of the browser and the container get closer together whether we use Chrome or Firefox, so the difference is speed-dependent.

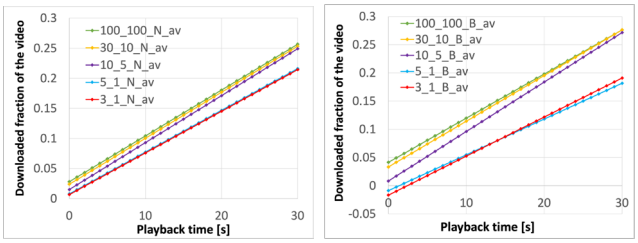
*3) Dependence of the reference client's parameters on the Internet access speed:* Since the most advanced video streaming providers are constantly sampling the maximum available speed of the available network connection, the behavior of the download controller they use changes as the available capacity begins to approach the source speed of the video stream from above. As mentioned above, the Chrome browser outperforms the container at higher speeds on Ubuntu Linux. However, as the speed decreases, this advantage decreases, then disappears and eventually becomes a disadvantage. A similar phenomenon can be experienced using Firefox. However, there




(a) Chrome (b) Firefox

Fig. 6: Browser and container characteristics (100/100 Mbps, Chrome/Ubuntu and Firefox/Ubuntu)

are much more minor differences between the cases, so for ease of interpretation, Fig. 7 shows the characteristics of the Chrome browser and the container at various access speeds (100/100, 30/10, 10/5, 5/1 and 3/1 Mbps, respectively). For transparency, the characteristics are now represented not by the delimiting lines but by their center line.




(a) Container (b) Browser

Fig. 7: Chrome browser and container characteristics on different speeds on Ubuntu

## VII. CONCLUSION

From an Internet provider's perspective, managing network resources is a crucial operational task, guaranteeing the quality of experience for cloud services that dominate their traffic mix. However, its endeavor to control resources, i.e., traffic engineering based on service popularity, may offend the original best-effort paradigm of the global Internet. This paper presented a real-time full-reference objective method to assess network neutrality with application awareness. The proposed method may support the Regulation (EU) 2015/2120 with scientific fundamentals. Furthermore, the expected scientific results can indirectly support EU citizens to objectively assess network quality by opening the way towards a new generation of network neutrality measurement tools for national communications authorities. The method incorporates three novelties to support the user-centric analysis of potential restraints affecting public services on the Internet: i) it supports application-specific measurements and involves real content and traffic, ii) the measured traffic originates from the content provider's cloud infrastructure, iii) the reference is created in real time. Accordingly, the paper also proposed a novel measurement layout. We have validated the incorporated client emulator and demonstrated the feasibility of the measurement

method with a video-on-demand use-case using an extensive set of laboratory measurements. In the future, we would like to extend the measurement capability of the implementation to a broader range of cloud services. Furthermore, the Hungarian National Media and Infocommunications Authority is evaluating the current version of the measurement system. It will potentially be available for the public to perform web-based network neutrality measurements.

## REFERENCES

[1] EU, "Regulation (eu) 2015/2120 of the european parliament and of the council," *Official Journal of EU L310*, 2015.

[2] BEREC, "Berec net neutrality regulatory assessment methodology," *BEREC BoR (17) 178*, 2017.

[3] BEREC, "Berec net neutrality measurement tool specification," *BEREC BoR (17) 179*, 2017.

[4] M. B. Tariq, M. Motiwala, and N. Feamster, "Nano: Network access neutrality observatory," Georgia Institute of Technology, Tech. Rep., 2008.

[5] G. Lu, Y. Chen, S. Birrer, F. E. Bustamante, and X. Li, "POPI: a user-level tool for inferring router packet forwarding priority," *IEEE/ACM Transactions on Networking*, vol. 18, no. 1, pp. 1–14, 2009. DOI: 10.1109/TNET.2009.2020799

[6] P. Kanuparthy and C. Dovrolis, "ShaperProbe: end-to-end detection of ISP traffic shaping using active methods," in *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pp. 473–482. DOI: 10.1145/2068816.2068860

[7] U. Weinsberg, A. Soule, and L. Massoulie, "Inferring traffic shaping and policy parameters using end host measurements," in *2011 Proceedings IEEE INFOCOM*. IEEE, 2011, pp. 151–155. DOI: 10.1109/INFCOM.2011.5934941

[8] Y. Zhang, Z. M. Mao, and M. Zhang, "Detecting traffic differentiation in backbone ISPs with NetPolice," in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement*, ser. IMC '09. New York, NY, USA: Association for Computing Machinery, Nov. 2009. ISBN 978-1-60558-771-4 pp. 103–115. DOI: 10.1145/1644893.1644905

[9] M. Dischinger, M. Marcon, S. Guha, P. K. Gummadi, R. Mahajan, and S. Saroiu, "Glasnost: Enabling End Users to Detect Traffic Differentiation." in *NSDI*, 2010, pp. 405–418.

[10] P. Kanuparthy and C. Dovrolis, "DiffProbe: Detecting ISP service discrimination," in *2010 Proceedings IEEE INFOCOM*. IEEE, 2010, pp. 1–9. DOI: 10.1109/INFCOM.2010.5461983

[11] A. Filastò and J. Appelbaum, "OONI: Open Observatory of Network Interference," in *FOCI*, 2012.

[12] Z. Zhang, O. Mara, and K. Argyraki, "Network neutrality inference," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4, pp. 63–74, 2014. DOI: 10.1145/2740070.2626308

[13] A. Maltinsky, R. Giladi, and Y. Shavitt, "On Network Neutrality Measurements," *ACM Transactions on Intelligent Systems and Technology*, vol. 8, no. 4, pp. 56:1–56:22, May 2017. DOI: 10.1145/3040966

[14] "WindRider - A Mobile Network Neutrality Monitoring System." [Online]. Available: https://users.cs.northwestern.edu/~ict992/mobile.htm (Accessed: 2021-02-28).

[15] X. Castoreo, P. Maillé, and B. Tuffin, "Weaknesses and Challenges of Network Neutrality Measurement Tools," in *2020 16th International Conference on Network and Service Management (CNSM)*, Nov. 2020, pp. 1–5. DOI: 10.23919/CNSM50824.2020.9269077

[16] E. W. Chan, X. Luo, and R. K. Chang, "A minimum-delay-difference method for mitigating cross-traffic impact on capacity measurement," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, ser. CoNEXT '09. New York, NY, USA: Association for Computing Machinery, Dec. 2009. ISBN 978-1-60558- 636-6 pp. 205–216. DOI: 10.1145/1658939.1658963

[17] M. Li, Y.-L. Wu, and C.-R. Chang, "Available bandwidth estimation for the network paths with multiple tight links and bursty traffic," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 353–367, Jan. 2013. DOI: 10.1016/j.jnca.2012.05.007

[18] ITU, "Mean opinion score (MOS) terminology," Recommendation ITU-T P. 800.1. *ITU-T Telecommunication Standardization Sector of ITU Geneva*, 2016.

[19] "YouTube Player API Reference for iframe Embeds," publication Title: Google Developers. [Online]. Available: https://developers.google.com/youtube/iframe_api_reference (Accessed: 2021-09-23).

**Péter Orosz** is an associate professor and the head of Smart Communications Laboratory at the Department of Telecommunications and Media Informatics, BME Hungary. He received his Computer Science master degree in the field of software engineering (2003) and Ph.D. in infocommunication systems (2010) at the University of Debrecen, Hungary. His research interest covers communication networks, network and service management, QoS-QoE managed networks, online QoE prediction for media services, and acceleration of network functions.

**Tamás Skopkó** is an assistant professor and a member of Smart Communications Laboratory at the Department of Telecommunications and Media Informatics, BME Hungary. He received his M.Sc in in field of software engineering (2002) and Ph.D in infocommunication systems (2017) at the University of Debrecen, Hungary. His research interest covers communication networks, measurement of service quality, online QoE prediction for media services, virtualization and softwarezation of network functions.

**Tamás Marosits** is an assistant professor and a member of Smart Communications Laboratory at the Department of Telecommunications and Media Informatics, BME Hungary. He received his M.Sc. (1996) and Ph.D. (2012) in the field of electrical engineering at the Budapest University of Technology and Economics, Hungary. His research interest covers communication networks, service quality measurement, online QoE prediction for media services, and statistical analysis of extremely large test result sets.

# Techno-economic analysis on Mobile Network Sharing contribution to social welfare at 4G-5G area in Hungary

Gábor Földes

***Abstract*—Telecommunication sector faces to parallel investments into both fixed and mobile (5G) networks, however return on investments lag behind profit expectations. Co-investment, like mobile network sharing is a cost efficiency enabler that may accelerate price decrease, may allow earlier, higher coverage and may improve capacity and quality parameters, like download speed, therefore altogether contributes to social welfare increase.**

**The purpose of this paper to assess the Hungarian mobile network sharing that not cleared by the competition regulator, however has been placed in unchanged form for 8 years.**

**The research question is to assess what is the connection between mobile network sharing and social welfare improvement at 4G - 5G mobile broadband rollout.**

**The finding is that, majority of network sharing procompetitive effects allowing benefits, but anticompetitive effects not causing marked distortion. Affordable connectivity prices for information society roots in operators' cost efficiency, however further research required to assess proper level of efficiency gains pass through to customers and appropriate level of access pricing to shared infrastructure for other rival operators in Hungary. Mobile Network Sharing's benefits may outweigh potential drawbacks, but due to lack of regulatory clearance, 5G rollout launched without sharing, causing social welfare loss.**

**The originality of the empirical research is despite network sharing not cleared, procompetitive advantages may outweigh anticompetitive ones.**

***Index Terms*—5G, mobile network sharing, cost efficiency, regulation, social welfare**

## I. INTRODUCTION

CORPORATE cooperation, like mobile network sharing has received increased attention across a number of disciplines in recent years. The EU aim is the widespread of 5G mobile broadband internet, however the optimal way forward is not clear. Corporates require more cooperation and even consolidation for higher economies of scale, however regulation put a stress on competition and innovation that expectations need to be harmonized.

This paper has been divided into five parts. *The first, introduction part* deals with the Mobile Network Sharing forms and the key driving factors from corporate and regulation sides. In the *second part, the literature review* proceeds the cooperation and co-investment forms, the telco sector as well as procompetitive and anticompetitive theories. *The third part* covers the research sub-questions, related measures and quantitative empirical research methodology. In *the fourth, discussion and results part* the Hungarian Network Sharing Agreement is introduced and key measurable procompetitive and anticompetitive effects are assessed. *The fifth, conclusion part* summing up the novelty finding that despite Network Sharing not cleared, the benefits might overweight competition concerns, and benefit loss is foreseen related to 5G rollout due to the absence of extended agreements.

One of the main cost efficiency measurements is the operator-operator collaboration, like horizontal agreements (Network Sharing), covered in this paper and TABLE I. shows the main dimensions of *Network Sharing Agreements (NSA)*.

TABLE I
OVERVIEW ON NSA TYPES

| Dimensions of RAN related Network Sharing Agreements | | |
|---|---|---|
| # | Dimensions | Sub-categories |
| 1. | Technology domains | ▪ passive infrastructure elements: towers, masts<br>▪ additionally active infrastructure elements, like radio –Multiple Operator Radio Access Network (MORAN)<br>▪ additionally spectrum – Multiple Operator Core Network (MOCN) |
| 2. | Spectrum bands | from a certain spectrum layer, e.g., 800 MHz to full scope |
| 3. | Technology generations | from a certain generation, e.g., 4G to full 2-5G range |
| 4. | Areas | ▪ density (rural, urban, exemptions, like capital)<br>▪ ratio of covered territory (geographic area)<br>▪ ratio of covered inhabitants<br>▪ the way of sharing, e.g., geographical split |
| 5. | Market context | ▪ number of operators (on the market; participating in NSA)<br>▪ position of participating operators (market shares)<br>▪ market concentration (HHI index) |

Gábor Földes, Vodafone Intelligent Solutions Europe / VoIS Hungary. (e-mail: gfoldes80@gmail.com). This is the journal version of paper [1] presented at ITS Europe 31. conference in Gothenburg, 2022.

DOI: 10.36244/ICJ.2023.1.9

| Dimensions of RAN related Network Sharing Agreements | | |
|---|---|---|
| # | Dimensions | Sub-categories |
| 6. | Operational model | ▪ cooperative model (e.g., Joint Ventures)<br>▪ contractual model (e.g., reciprocal, one-way sharing)<br>▪ transfer pricing (public, cost based versus any other cases) |
| 7. | Duration | Temporary or mid/long term permanent |
| 8. | Driver | Commercial benefit or regulatory imposed |

Source: [1]

Based on regulatory practice, shared technology domains (passive-active) and covered areas (rural-urban) are the most decisive dimensions from the list above. In practice the main cases handled by regulation are *passive sharing* and *active sharing* (MORAN), but additional spectrum sharing (MOCN) many cases ruled-out. The higher the scope, the higher is the financial benefit (saving) and parallel the highest the potential restriction impact on competition and innovation incentives. The higher scope preferred by Mobile Network Operators (MNOs), the smaller scope preferred by regulators. The expected network sharing's cost savings may exceed one third of investment capital expenditure (CAPEX) and operational expenditure (OPEX) based on Arthur D. Little consultancy estimation [2].

## II. LITERATURE REVIEW

At first, I take the focus to literature on cooperation forms, review of telecommunication value-chain and market structure. Secondly, I show the NSA relevant legal framework from sector and competition regulation point of view. Thirdly I summarize recent studies related to NSA benefits (procompetitive aspects) and drawbacks (anticompetitive concerns).

### A. *Cooperation forms*

Mobile Network Sharing Agreements from competition policy point of view belongs to merger and horizontal agreement categories. From industrial organization aspects the literature refers to it under coopetition definition. The telecommunication sector regulation refers for the cooperation as co-investment.

*Mergers and horizontal agreements*
A *merger* is an agreement between two existing companies to form a new legal entity under one corporate name. The *horizontal merger* is a merger between competitors. Based on Motta [3], the merger has to be evaluated whether the merged firm can unilaterally (non-coordinated) exercise market power and raise prices, as well as whether can apply pro-collusive (coordinated) actions that might substantially lessen the competition and raise prices. The *horizontal agreement* is a softer form of horizontal merger, the NSA is handled as *a horizontal production agreement.*

*Coopetition*
From industrial organization approach Brandenburg-Nalebuff [4] name NSA-like cases as a *coopetition,* a parallel cooperation and competition at the same time. It is a cooperation in creating value, and a competition in dividing it up simultaneously. In NSA there is a technology production (mobile network services) cooperation and there is fierce competition at end-user retail mobile telecommunication market.

*Co-investment*
The new European Electronic Communications Code ("Code" or EECC) introduced in 2018 [5] the *co-investment* for VHCN, that is an exchange for regulatory flexibility compared to standard access remedies. Originally it is used for fixed network investments, but mobile network sharing among competitors is also a typical co-investment.

*European MNOs standpoints related to cooperation*
European MNOs requesting room for building scale in the telecom industry [6]. Vodafone CEO also added the European mobile market is *"hyper fragmented",* and not only *"cooperation"* (interpreted: network sharing), but also *"consolidation"* (interpreted: merger) needed.

*Cooperation levels related to network sharing types*
In case of infrastructure owner MNOs, Fig. 1 shows the different sharing status.

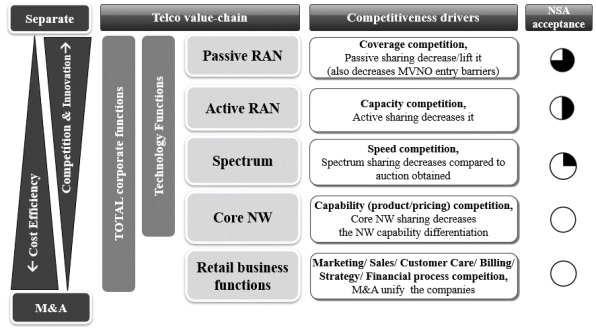



Fig. 1: Overview on NSA related horizontal agreement milestones to M&A

Source: [7]

The higher the horizontal agreement scope (covered network domains and corporate functions), the closer the horizontal merger and acquisition (M&A) case that represents the highest cost efficiency but may soften competition. NWS, as a network production agreement is on the mid way, affecting just coverage and/or capacity competitiveness drivers, while product capability in core network and all other business function differentiators are unimpacted. Regulatory acceptance is positive and high for passive sharing, but gradually decreasing involving active elements or even spectrum. Further co-investment covering whole network functions or even other corporate functions, would reach already the status of acquisition, exceeding the network sharing framework.

*Telecommunication value-chain and market structure*
Pápai [8] identified the following three layers, as production layer, wholesale layer and retail layer. Production layer

contains the network technology as a service, which is sold by wholesale either to own retail internal end customers or to Mobile Virtual Network Operators (MVNO). Therefore, the network technology market becomes a functional *upstream market* and the retail market is a *downstream market.* Mobile Network Sharing Agreements impact the production upstream market only.

Network technology production in upstream market consists of Radio Access Network (*RAN* – e.g., base station, towers), Transmission (TRM, mainly has an access part and core part) and *Core Network*, which is the intelligent part of the network. Mobile Network Sharing Agreements impact only the RAN and the access TRM parts. Important note that only RAN production part of network technology upstream market is affected, the Core part remains fully independent that allows MNOs to determine and differentiate their own product, service and pricing capabilities.

The *relevant market* is where from supply aspect the substitution is limited, and from demand side it is without choice if a certain type of service is needed. In this research paper the relevant market is the retail telecommunication service market at HU level.

*B. Regulatory framework*

The regulation has dual aims: on one side encourages investments and higher coverage, on the other side preserves incentives for competition and innovation.

*Sector regulation*

The Sector regulation (National Regulatory Authorities – NRAs), represented by Body of European Regulators for Electronic Communication (BEREC) at EU level, has equal focus on meeting Digital Decade 2030 connectivity (100% of EU households covered by Gigabit network and 100% of population covered by 5G) in a cost-efficient way, as well as maintain incentives for preferred infrastructure-based competition and innovation.

The direct regulation elements are the OECD guideline on Wireless Market Structures and Network Sharing [9] and BEREC Common position on infrastructure sharing [10]. The OECD report makes use of competition policy categorizations of potential anticompetitive behavior and distinguishes between potential unilateral and coordinated effects of infrastructure sharing agreements. The BEREC common position from 2019 principally assesses the feasibility of infrastructure-based competition and refers to the population density of the areas.

*Competition regulation*

The competition regulation (National Competition Authorities – NCAs) main focus is to prohibit such cooperation that may prevent, restrict or distort the competition, that may decrease social welfare. However, it handles exemption, where the restriction of competition kept at minimum level and compensated by technical or economic benefits, from which fair share pass through to customers. In other words, cost efficiency benefits considered just on those cases, where it outweighs potential competition drawbacks. [11]

*C. Mobile network sharing related procompetitive and anticompetitive effects*

In the last part of literature review the Mobile Network Sharing Agreement relevant main benefits, drawbacks and a potential netting are summarized. Recommendations for more likely regulatory approvals will be concluded also.

*Procompetitive effects (benefits)*

NSA related benefits for social welfare are listed in TABLE II.

TABLE II
OVERVIEW ON NSA PROCOMPETITIVE EFFECTS

| # | Benefit | Main elements |
|---|---|---|
| 1. | Improved Network Efficiency | Sharing leads to faster and wider rollout of high-speed networks coverage and higher quality. |
| 2. | Cost reduction and efficiency gain pass through consumers | Sharing of deployment costs, sharing of operational costs, leading to lower prices. |
| 3. | Enhanced customer choice | Enhanced competition, benefiting customers in terms of lower prices and facilitate entry for third-party operators. |
| 4. | Environmental benefits | Single infrastructure, with less impact on environment. |

Source: Author' s summary on [9], [10], [11]

*Anticompetitive effects (drawbacks)*

The discussion on potential anticompetitive effects of NSA follows the structure of BEREC Common Position and OECD guideline, summarized in TABLE III.

TABLE III
OVERVIEW ON NSA ANTICOMPETITIVE EFFECTS

| # | Drawback | Main elements |
|---|---|---|
| *Unilateral (non-coordinated) effects* | | |
| 1. | Reduced incentive for investment | Shorten incentives for unilateral investment or hold-up strategies resulting in lower total network investment. |
| 2. | Decreased differentiation | Constrained for differentiation from co-investment partner, unilateral investment, service innovation and other independent strategies; |
| 3. | Excessive access or transfer pricing | Raising internal and/or external access prices to soften competition and raise final prices at the downstream (retail) market |

| # | Drawback | Main elements |
|---|---|---|
| *Unilateral (non-coordinated) effects* | | |
| 4. | Incentive for foreclosure | Raise market entry barriers by excluding third-party potential rivals with deterring access conditions (e.g., prohibitive, excessive access prices, too high commitments). In case of MVNOs, it is vertical impact also. |
| 5. | Cost disadvantages | Outlayer competitor might face to a cost disadvantage, weakening its competitive position, (but any investment in coverage or quality has this effect, is a normal feature of competitive dynamics) |
| 6. | Deter late co-investments | Cherry-pick from third parties, deter late co-investments. |
| *Pro-collusive (coordinated) effects* | | |
| 7. | Explicit collusion | Potential explicit collusion, depending on the possibilities to exchange information and to coordinate on more than joint investment required. |
| 8. | Tacit collusion | Potential tacit collusion depending on partners' ability to: coordinate on a collusive 'agreement', detect deviations from the agreement and punish deviations. |
| *General effect* | | |
| 9. | Network supply security | Network resilience: a single rather than multiple infrastructures may imply that in case of network fault, consumers cannot switch to another network. |

Source: Author' s summary on [8], [9], [10]

*Net impact review of pro- and anticompetitive effects*

Motta and Tarantino [13] studied the impact of a network sharing agreement between two (mobile) operators on prices and investment in an oligopolistic mobile market. They analyzed the incentives to invest in cost-reducing innovations: coordinate their investment while competing in prices. They found that without merger's investment synergies, a horizontal merger reduces consumer welfare. If synergies are sufficiently large, consumers benefit, as network sharing agreement leads to lower prices and higher investment, and therefore increases consumer surplus.

## III. Research Methodology

The *research aim* of this paper to quantify, measure and assess mobile network sharing related procompetitive and anticompetitive effects, listed in TABLE IV.

TABLE IV
Overview on NSA effects related Measures

| # | Benefit/Drawback | Measures |
|---|---|---|
| Procompetitive effects assessment | | |
| 1. | Improved Network Efficiency | 4G and 5G related: Coverage, Penetration, Used Capacity (monthly average data traffic per subscriber), Speed |
| 2. | Cost reduction and efficiency gain pass through consumers | Unit cost decrease and fair share pass through to retail prices (bucket view) |
| Anticompetitive effects assessment | | |
| 3. | Excessive access or transfer pricing, incentive for foreclosure | Cost based and open transfer prices, 3rd party partnerships |
| 4. | Market context and cost disadvantages | Market context impact: market concentration, market share (voice subscriber, data traffic and service revenue based). Revenue based profitability (EBITDA/Sales) |
| 5. | Decreased incentive for competition (differentiation), incentive for collusion | Different market strategies and offers |
| 6. | Reduced incentive for investment | CAPEX intensity (CAPEX/Sales) – limited accessibility |

Source: [14]

In the fourth, discussion and results section these measures are evaluated for Hungary (HU) for 2014-2021 period.

*This study employs quantitative empirical data based approach* both for primary and secondary data analysis. The research design contains conclusive researches, in both descriptive and causal approaches. Descriptive research with cross-sectional design and in some cases also with longitudinal design applied for all measures related to network quality and market context description. Causal research is limited to representative sample analysis, as not enough reliable data (observations) are available for correlation analysis of profitability and investment intensity neither for cross-sectional, nor for longitudinal design.

## IV. Discussion and results of Hungarian Mobile Network Sharing assessment

Although competition regulation raised potential serious concerns already in 2015, the case have not yet been concluded, therefore NSA has been in place in unchanged form for 8 years.

TABLE V. shows the main characteristics of HU mobile market.

TABLE V
OVERVIEW ON HU MOBILE MARKET

| Dimensions (data for 2021 YE) | Hungary |
|---|---|
| Territory | 93,030 sq KM |
| Population | 9.73 Million |
| GDP per capita (PPP based, in USD), 2020 YE *(EU27 avg: 44,791 USD – Worldbank)* | 33,076 USD |
| Total subscribers (active SIM cards) (HU: 2021 H1) | 11.23 Million |
| TOP 3 MNOs and Market share (subscriber based, active SIM) (HU: 2021 H1) | 1. Magyar Telekom (DT): 43.9%<br>2. Vodafone: 27.4%<br>3. Telenor/Yettel (PPF): 26.8% |
| Avg. Voice minutes (per active SIM, monthly) *(EU27 avg: 186 min – Statista.com)* | 200 min |
| Avg. Data traffic (per data using active SIM, monthly) (HU: 2021 H1) *(CEE: 9.9 GB; Western EU: 15.5 GB – Ericsson)* | 7.2 GB |

Source: Author's summary based on [15], [16]

HU market is a typical 3 MNO player market, MVNO presence become insignificant, the market is matured, however slight changes took place, in HU Vodafone has grabbed the 2nd position based on more measures in recent years.

*Mobile network sharing agreement in HU*

In the Hungarian case the integrated incumbent market leader (both fixed and mobile) Magyar Telekom (subsidiary of Deutsche Telekom) entered into an MOCN (passive + active asset and spectrum sharing) NSA with second market player Telenor Hungary (at that time owned by Telenor Group, later sold to PPF Group) on 4G LTE 800MHz rollout in the whole country with geo split operation, except capital Budapest. The highest cost saving potential, but from regulatory aspects most opposed MOCN NSA concept might came from Telenor side related to Danish market, where Telenor Denmark and Telia Denmark had an approved MOCN NSA from 2012. Third market player Vodafone appealed the NSA at NCA. Local NRA approved the agreement driven by technology spectrum efficiency aspects, however local NCA raised concerns in 2015, but final decision not made till nowadays. This resulted, that no further NSA made in the market, however 5G non-standalone rollout started with 2-4G lifecycle network swap in separated 3 networks, apart from the only shared 4G L800 MHz layer.

TABLE VI. summarize the main characteristics of the HU NSA.

TABLE VI
OVERVIEW ON HU NSA

| # | Dimensions | Hungary |
|---|---|---|
| 1. | Technology domains | MOCN (passive + active asset + spectrum sharing) |
| 2. | Spectrum bands | 4G 800 MHz – rural coverage spectrum (temporary 3G 900MHz - rural coverage spectrum) |
| 3. | Technology generations | 4G (temporary 3G), no 5G extension |
| 4. | Areas | Population coverage: 80%<br>Geo split: Magyar Telekom (East), Telenor (West), exemption Budapest |
| 5. | Market impact | Started: from 2014 (4G 800MHz), 2016-2019 temporary for 3G (900MHz) rural<br>3 MNO market, HHI is: 3517<br>Parties: Magyar Telekom (No.1 player) and Telenor (No.2 player) |
| 6. | Operational model | contractual (4G reciprocal; temporary 3G one-way sharing) |
| 7. | Duration | from 2014, end 2029 (expires the 800MHz license) (temporary only 3 years till 2019)<br>transfer price: no public information disclosed |
| 8. | Driver | commercial driven |

Source: [1]

*Regulatory assessment*

*In 2015 NMHH, the Hungarian NRA* was notified by Magyar Telekom and Telenor Hungary about the NSA to mutually and partially share the spectrum with each other in the 4G LTE 800 MHz band nationwide with exception of capital Budapest. NMHH approved the lease, as a secondary trading. NMHH declared, the agreement enabled both operators to offer a larger capacity and better technology characteristics. NMHH examined that the individual obligations linked to the individual licenses are fulfilled. As NMHH had no competence to examine the competition law aspects of the lease agreement, NMHH sent its decision to the NCA for information. [17]

*In 2015 GVH, the Hungarian NCA* launched the investigation process with respect to the NSA.

In 2018 GVH held unannounced inspections at both parties of the cooperation in order to examine whether there was collusion during the spectrum tender in 2014. Therefore, NSA related investigation was extended for last spectrum tender, contained 800 MHz. The aim of the investigation was to assess whether the agreement may have been aimed at sharing the relevant market in advance and at coordinating the bids of the undertakings with the purpose of fulfilling the tender's

conditions. These practices may have violated provisions of the Hungarian Competition Act and TFEU, which prohibit agreements restricting competition. [18]
In 2019 GVH communicated that investigation still ongoing. Probably the Hungarian NRA also wait for the EC final decision related the Czech NSA case. Despite Czech case already reached an agreement in July 2022, HU case still uncleared.

In this chapter at the research methodology listed potential procompetitive and anticompetitive measures will be analyzed for concrete HU market figures in order to assess the NSA impact.

*A. Network Efficiency Improvement*

*Coverage*
The cost driver of telecommunication investments moved from voice to data traffic, therefore the expected NSA benefit can be measured related to 4G LTE mobile broadband internet, shown in TABLE VII.

TABLE VII
OVERVIEW ON HU 4G COVERAGE SURPLUS

| 4G coverage (% of households [average of operators]) | 2018 | 2019 |
|---|---|---|
| EU | 94% | 96% |
| Hungary | 96% | 97% |

Source: Author' s summary based on [19], [20]

TABLE VII. displays that HU had higher coverage in the early matured period of 4G, compared to EU average. It means that the NSA boosted 4G rollout was faster, than in other EU countries.

*Capacity (Speed and Quality)*
Capacity drives the download and upload speeds of the network, from which the ranking positions out of 87 countries displayed in Fig. 3.

| Country (Ranking #) | 4G availability | Video experience | Download speed | Upload speed | Latency |
|---|---|---|---|---|---|
| Hungary | 8. | 2. | 11. | 8. | 2. |

Fig. 3. Overview on 4G download speed experience in Q1 2019

Source: [21]

The download speeds are in the high segment for Hungary, to which 4G NSA may contributed also. The Hungarian MOCN NSA (network sharing includes spectrum sharing) also enabled the good results. NSA participant Magyar Telekom and Telenor in the 800 MHz spectrum tender in 2014 successfully acquired 10-10 MHz blocks unilaterally that could combine to double (20 MHz) bandwidth.

*Mobile Data usage (GB traffic)*
Data usage trends are summarized in Fig. 4 and 5.

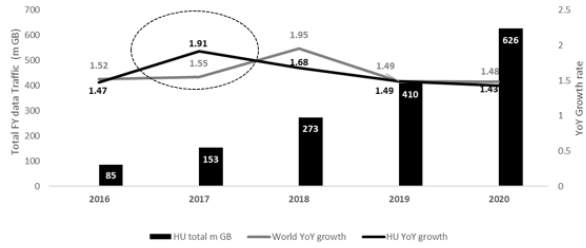



Fig. 4. Overview on Total Data traffic growth in HU

Source: Author' s summary based on [15], [16]

In HU after major part of the 4G LTE network on 800 MHz shared rolled out (2016-2017), there was a tremendous year-on-year total data traffic increase (almost doubled the traffic) in 2017, which peak was 1 year ahead of data blow-up at worldwide. In case of HU, the NSA led faster rollout was the definite driver of data growth and it enabled Telenor to launch a differentiating unlimited data offer.

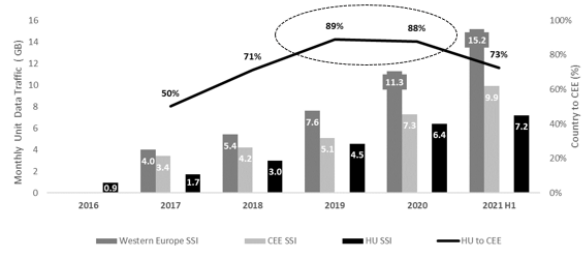



Fig. 5. Overview on Monthly Unit Data traffic in HU

Source: Author's summary based on [15], [16]

After major part of shared 4G LTE network rolled out, the GAP between CEE and HU SIM card level unit data traffic began to narrowing and NSA enabled to reach 89% of the CEE average traffic in 2018 due to the mentioned Telenor offer. However later the lack of shared 5G rollout also contributed to the data growth slowdown and the GAP started widening again.
*The findings* clearly indicate that NSA contributed to network efficiency improvement and increased social welfare in HU.

*B. Cost reduction and efficiency gain fair sharing*

Network OPEX (site rental, energy cost, operational & maintenance cost, personnel cost, transmission cost, spectrum yearly fees) were responsible for 48% of telco Total OPEX in 2021 based on Analysys Mason [22] and the dominant part related to RAN. The 30-40% NSA related efficiency gain results some 10% cost savings at company level.
GB costs, therefore the GB prices also decreasing without Network sharing, as data traffic growth exceed the cost increase. Price decrease coming from a mixture of new technology generation and additional cost efficiency actions, like network sharing.

*Retail price benchmarking*

In the EU one of the most accepted telco price benchmarks is the *Empirica report* that key findings on HU usage pattern fitting baskets summarized in TABLE VIII.

TABLE VIII
OVERVIEW ON HU USAGE PATTERNS AND EMPIRICA BASKETS

| HU Baskets Data in 2020 | Voice (call unit) | Voice (minutes) | Data (GB) |
|---|---|---|---|
| HU SSI | 70 | 243,9 | 6,4 |
| HU LSI (w FWA) | 0 | 0 | 28,3 |
| Empirica I6 basket (voice and data - SSI) | 30 | 80 | 5 |
| Empirica I4 basket (voice and data - SSI) | 300 | 600 | 5 |
| Empirica MBB5 basket (data only - LSI) | 0 | 0 | 20 |

Source: Author' s summary based on [15], [24]

Fig. 6. shows the I6 and MBB5 basket relevant overall detailed ranking.

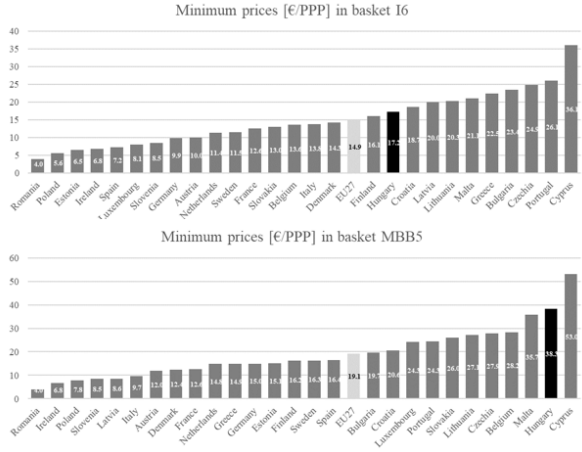



Fig. 6. Detailed country price ranking in I6 and MBB5 baskets

Source: [24]

Based on Fig. 6. HU market belongs to the (most) expensive markets in voice and data bundle (SSI - Small Screen Internet via smartphone) and data only (LSI - Large Screen Internet via stick) most relevant baskets, covering average usage habits. HU position is better in bundled I6 package, but almost at the high-end in data only segment.

Other standard reports show quite similar picture also based on *OECD Mobile voice and Data as well as the Mobile Broadband reports* made by Teligen Strategy analytics or *Rewheel researcher*.

*There is a difficulty in reaching a final conclusion* on whether fair share from NSA related efficiency gains passed through to customers, despite realized cost reduction.

*C. Excessive access or transfer pricing*

Limited information published on charging mechanism between NSA parties to judge *excessive access or transfer pricing*, as it belongs to confidential information on commercial contract terms. It is known that HU NSA not built on cooperative agreements (special join venture), but on contractual agreement (reciprocal access sharing).

Excessive access prices may *incentivize foreclosure of other MNOs or MVNOs.* Vodafone was the biggest outlayer telco in HU. When Vodafone realized financial disadvantages of outlayer position, it would have been difficult go for nationwide asset company model to serve three MNOs both from corporate and regulation perspective.

Regarding other players in the HU market MVNOs continuously loosed ground as second brands were retired (Tesco, Lidl Mobile) or acquired (UPC mobile by Vodafone-Liberty transaction). There is one half MNO, half MVNO player, the ex-subsidiary of Romanian Digi, bought by 4iG in 2021. Digi managed to buy only very limited spectrum at 1800 MHz and 3600 MHz in the mobile segment, therefore it was unable to launch a fully-fledged MNO service. Digi made a lot of local deals for passive sharing even with the NSA parties, however such a big deal that was disclosed as in Germany between 1&1 Drillisch and Vantage TowerCO, was not announced. 4iG resolved the lack of spectrum in mobile services by buying the Vodafone HU in 2023.

*D. Market context and cost disadvantages*

*Market concentration*

Bourreau in [23] provided a market concentration overview for the European market per country. Calculations based on GSMA and company data, applying the cumulative criteria of number of network operators (3 or 4) and the industry concentration (measured by Herfindahl-Hirschman Index, HHI). Out of EU 27 markets only 9 markets are 4-player and 18 markets are 3-player. Out of 3 player markets, HU located in the mid-low segment in HHI ranking with value of 3517, that means there are 13 Member States with a higher industry concentration among 3-MNO market. So, there is no extraordinary market context that would restrict market competition in HU market.

*Market shares*

Development of market shares may reveal unfair market behavior and anticompetitive trends, if NSA outlayer companies loosing continuously ground. HU relevant market share (MSH) trends are displayed on the followings: Fig. 7. shows the active (voice) SIM and service revenue based, Fig. 8. the data generating SIM based and data traffic, GB based.

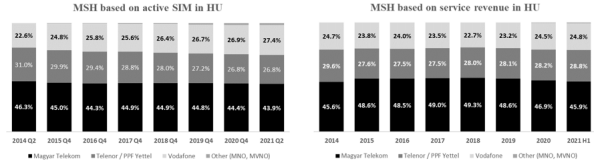



Fig. 7. Market shares based on active SIM cards and service revenue in HU

Source: Author's own summary based on [15]

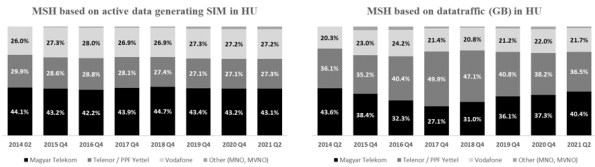



Fig. 8. Market shares based on data generating SIM cards and data traffic in HU

Source: Author's own summary based on [15]

It is visible that NSA outlayer Vodafone's MSH in all of the 4 cases has increased in HU that is significant finding refers to NSA not distorted the market position of Vodafone, therefore the competition. The share of other MNVOs, MNOs, also show some slight increase in the majority of the categories. The highest increase in Vodafone's market share took place in active SIM based comparison in HU, where almost 5% increase was enough to overtake NSA participant Telenor in the ranking and grab the 2nd position (Fig. 8.).

Summing up, the NSAs in HU not restricted the competition in terms of outlayer operator loosing market share. This is an important evidence that the opposed NSAs did not restrict the competition and not cause a harm to social welfare.

*Cost disadvantages*

Cost disadvantage as an unilateral effect may restrict competition via weakening competitive position of NSA outlayer companies. The relevant indicator is the *EBITDA/Sales* revenue based profitability margin, presented in TABLE IX.

TABLE IX
OVERVIEW ON EBITDA MARGINS IN 2019 IN HU

| EBITDA /Sales margin (2019FY, After Lease - IFRS16 corrected back) | HU |
|---|---|
| Telekom (Magyar Telekom – converged) | 29.4% |
| PPF (Telenor HU – pure mobile) | 36.7% |
| Vodafone (Vodafone HU both pure mobile) | 22.3% |

*Source: Author' s summary based on operators' Financial reports for 2019*

In Hungary the NSA outlayer Vodafone's EBITDA margin lagged behind significantly and stood only at 22.3%. Vodafone's market share after more than 15 years entering into the HU market just slightly exceeded 22% on active SIM cards. On one hand Vodafone's cost structure and amount is similar to competitors, who own 30% - 40% of the market revenues, therefore gaining higher profitability. On the other hand, the analyzed 2014-2021 period was the time, when Vodafone managed to overtake Telenor and grab 2nd position in the market gradually. Based on Vodafone's strategy the first priority was to increase SIM volume and revenue market shares, therefore overspent the market, thus EBITDA margin stuck in under 25%. So potential cost disadvantage of outlayer NSA position was not the driver of lowest profitability.

The other financial indicator of potential cost disadvantage is *CAPEX intensity (CAPEX/Sales margin)*. In 4G investments' peak 2-3 years period, it may have resulted marked extra investment for outlayer Vodafone, but not such a tremendous that could weaken its competitive position, shorten market competition and harming social welfare.

*E. Decreasing incentive for competition (differentiation) and investment*

At theoretical level in section II.A Cooperation forms, it was presented that although passive NSA in coverage and active NSA in capacity may decrease differentiation in RAN segment, for whole network capability in the Core NW segment is responsible for product and service differentiation. Core segment is unshared and all of the other business (marketing, sales, customer care) and overhead (strategy, communication) functions of the participants are separately managed.

In spite of NSA, preserved differentiation in Network can be demonstrated at the Hungarian market by one case study. In 2016, after majority of the 4G LTE 800MHz network was rolled-out, the pure mobile operator Telenor Hungary launched an out-of-box reverse (data first) unlimited data offer [25]. The enabler was the NSA with more and faster rollout, and as a pure mobile operator wanted to immediately take and advantage of 4G countrywide (coverage and capacity). As Fig. 8. showed, Telenor's data traffic blowup at that. It is highly demonstrative example, that NSA participants on the same network how different business strategy can build.

Reduced differentiation might rooted in *wider information exchange*. Considering differentiation examples above, there are no sign of *tacit collusion*. However, all current full-fledged MNOs in HU market are highly incentivized to preserve the total market profitability and therefore in competition they do not introduce such disruptive and status quo changing offers that might generate a negative spillover effect for total market profitability.

*CAPEX/Sales margin* is the industrywide measure of investments, however there is limitation on availability. Based on financial reports Magyar Telekom converged operator has stable CAPEX/Sales ratio, exceeding 15% without spectrum license acquisition cost and After Lease (IFRS 16 correction back) effect, which is in line with industry investments. PPF Telecom Group, including converged and pure mobile operators as well as asset company, for the CEE region reported an overall CAPEX/Sales margin under 15%, exceeding 12%, as blended figure of higher converged and lower pure mobile investments. For Vodafone no local data reported. Based on available information there is no sign of reduced investments due to NSA.

## V. CONCLUSION FOR 4G, 5G AND SOCIAL WELFARE

The main goal of this study was to assess the Hungarian mobile network sharing agreement impact for social welfare.

MNOs strive for more cooperation (horizontal agreement) and even consolidation (merger) in the "hyper fragmented"

European market to reach better economies of scale, productivity and cost efficiency. Mobile Network sharing, as a horizontal agreement may restrict much less the competition, than a full-scale merger, however both sector and competition regulation has condition lists for approvals.
The more stricter competition policy lays on TFEU, that on one hand prohibits competition distorting or preventing agreements, however also allows an exemption where the restriction is kept to the minimum level and compensated by technical or economic progress, and a fair share of the efficiency gains passed to customers. Effects summarized in TABLE X.

TABLE X
OVERVIEW ON NSA EFFECTS AND FINDINGS IN HU

| # | Benefit/Drawback | Measures | HU |
|---|---|---|---|
| Procompetitive effects assessment | | | |
| 1. | Improved Network Efficiency | Coverage, penetration, capacity, speed | ✓ ● |
| 2. | Cost reduction and efficiency gain pass through consumers | Unit cost, retail prices | ? ◑ |
| Anticompetitive effects assessment | | | |
| 3. | Excessive access or transfer pricing, incentive for foreclosure | Cost, 3rd player partnership | ? ◑ |
| 4. | Market context and cost disadvantages | Market concentration, market share, EBITDA/Sales | ✓ ◕ |
| 5. | Decreased incentive for competition (differentiation), incentive for collusion | Different market strategies | ✓ ◕ |
| 6. | Reduced incentive for investment | CAPEX/Sales | ✓ ● |

*Source: Author's own summary*

*Conclusion for 4G*

*The conclusion* is that there is no doubt on *network quality* related benefits of NSA and there is no visible investment decrease. The assessment is still positive on *market context* with the only exemption that 1st and 2nd players of the market made the NSA. Market concentration, market share trends do not justify shortening competition and the profitability trends do not display a serious cost disadvantage for outlayer players. There is no absence of *differentiation* in retail perceived strategies (rooted also in RAN segment), so there is no reduced incentives for competition. There is no sign of *tacit collusion,* however there are also no market status quo changing disruptive offers that may jeopardies market profitability. The assessment is unclear related to *efficiency gain pass through customers,* as based on standard benchmarks there is no clear evidence on relative improvement of retail prices compared to other EU countries, however it is doubtless that unit GB prices are falling. There is no transparency on *access or transfer prices* and despite no direct connection can be detected, MVNOs' presence is low and no countrywide access deal has been published.

*The conclusion for net impact* is that there is no critical restriction for competition and harm for social welfare, therefore anticompetitive impacts kept at quite low level that are overweighted by technical and economic progress by efficiency gain partial sharing.
*The main finding and the originality of this study,* that despite active asset (MORAN) and spectrum (MOCN) sharing NSAs are highly opposed, the benefits outweigh potential drawbacks, the net contribution for social welfare is positive.

*Consequences for 5G*

Although 4G shared rollout not cleared by the regulator, but allowed to remain in operation in unchanged form, the concrete harm of uncertain regulation realized in 5G with more costly separate rollouts. Unfortunately, in HU market the NSA has not evolved further, as MNOs might evaluated at high risk to expand the agreement for 5G until existing NSAs are not cleared. This caused two critical problems from cost efficiency point of view: (1) 5G spectrum tendering (700MHz, 3500MHz) and 5G rollout was the current window of opportunity to extend NSAs for 5G; (2) around 2020 MNOs met the need of parallel 5G rollout and 2-3-4G RAN lifecycle swap that could have enlarged the savings potential for 2-3-4-5G scope in RAN consolidation. The missed savings opportunity not only a loss for MNOs, but has a cost for society as may slow down data unit price decrease, impacting 5G rollout and penetration development.

GSMA disclosed a study in 2019 on expected 5G cost evolution, that stated the TCO of 5G rollout is higher, than for 4G. GSMA said that the 5G baseline cost at TCO level is by 46% higher than the 4G reference cost, that could be optimized by network sharing and virtualization opportunities to reduce increase to 24% [26].
In Hungary the eNET consultancy prepared the National 5G strategy update, in which the bottom-up modelled assumption was, that the operation (OPEX) of 2-3-4-5G network in 2027 could be 1.7 times higher, than the 2-3-4G was in 2019, and the investment (CAPEX) for 2020-27 could be 2.5 times higher, than was in 2013-2019 without virtualization and further network sharing. The assumption for TCO level savings potential was almost 20% with passive sharing and top on came more than 10% additional for active sharing. [27]

*Conclusion for 5G*

There is trade-off between cost efficiency as well as competition and innovation. Due to lack of extended NSAs, substantial cost savings are missed, therefore cost efficiency was ranked into second position, after competition and innovation. *Final conclusion extended for 5G* in HU is that, on one hand there is a clear missed cost savings opportunity from 5G shared rollout, that may slowdown unit prices decrease and 5G widespread, on the other hand there is an unclosed evaluation of potential competition restrictions of 4G sharing, that's potential net negative impact based on the findings of this study could not be justified.

Due to data monetization challenges and missing savings incentives from extended NSAs the 5G rollout currently had a slower pace based on DESI report: 5G population coverage was 7% in HU, compared to EU average 14% in 2020. [28]

*Future scope*

The 5G is a technology shift, corresponding with *network function virtualization (NFV), software defined networks (SDN)* enabling *open RAN* multivendor concept. At the second part of 5G lifetime the rollout will move *from macro cells to micro cells* in high density urban areas that requires extra investments and need of cooperation. However NSAs under current regime, especially active sharing, in particularly in the high-density populated urban areas are considered potentially competition restrictive, therefore harmful for social welfare, compared to infrastructure-based competition. The virtualization (NFV, SDN) and open RAN may move the competition into software based segment and increase the competition landscape in upstream vendor market that may contribute to more differentiation in RAN segment and more intensive competition in downstream market. Therefore *the current competition approach may require a full reassessment related to mobile network sharing in high density 5G macro and micro cell areas* that might be the future scope of next research.

## REFERENCES

[1] G. Földes, "Financial and regulatory assessment of Mobile Network Sharing as a trigger of cost efficient 5G rollout in CEE." 31st, *ITS Europe 2022 Conference*. 2022. [Online] Available: https://www.econstor.eu/handle/10419/265626, DOI: 10.2139/ssrn.4189476

[2] Arthur D. Little, "Network Sharing in the 5G Area." By Taga, Karim, Arthur D. Little Consultancy, 2020. [Online] Available: https://www.adlittle.com/sites/default/files/reports/adl_network_sharing_5g_era.pdf

[3] M. Motta, "Competition policy, Theory and Practice." *Cambridge University Press*, 2003.

[4] A. M. Brandenburg, B. J. Nalebuff, "Coopetition" *Crown/Archetype publisher*. 1997.

[5] EECC, "Directive (EU) 2018/1972 of the European Parliament and of the Council as of 11.10 December 2018 establishing the European Electronic Communications Code." European Commission, 2018. [Online] Available: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018L1972

[6] ETNO, "Joint CEO statement: Europe needs to translate its digital ambitions into concrete actions." *ETNO*. 2021. [Online] Available: https://www.etno.eu/news/all-news/717-ceo-statement-2021.html

[7] G. Földes, "The cost savings options for sustainable 5G." *HTE (Scientific Association for Infocommunications in Hungary)*, 2021. [Online] Available: https://www.hte.hu/documents/10180/4687623/5G_NWS_voRAN_Financials_HTE_20210316_v2.pdf

[8] Z. Pápai, G. Csorba, P. Nagy, A. Mclean, "Competition Policy Issues in Mobile Network Sharing: A European Perspective." *Journal of European Competition Law & Practice*, vol. 11, Issue 7, pp. 346–359, 2020, [Online] Available: DOI: 10.1093/jeclap/lpaa074

[9] OECD, "Working Party on Communication Infrastructures and Services Policy." Wireless Market Structures and Network Sharing." 2014. [Online] Available: https://www.oecd.org/officialdocuments/publicdisplaydocumentpdf/?cote=DSTI/ICCP/CISP(2014)2/FINAL&docLanguage=En

[10] BEREC, "BEREC Common position on infrastructure sharing." *BEREC*, 2019. [Online] Available: https://berec.europa.eu/eng/document_register/subject_matter/berec/regulatory_best_practices/common_approaches_positions/8605-berec-common-position-on-infrastructure-sharing

[11] TFEU, "European Commission guideline. Guidelines on the applicability of Article 101 of the Treaty on the Functioning of the European Union to horizontal co-operation agreements. "*European Commission*" 2011. [Online] Available: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52011XC0114(04)&from=E

[12] M. Motta, E. Tarantino, "The effect of horizontal mergers, when firms compete in prices and investments" Economics Working Papers 1579, Department of Economics and Business, Universitat Pompeu Fabra. 2017. [Online] Available: http://www.eief.it/files/2017/09/motta.pdf

[13] M. Bourreau, S. Hoernig, W. Maxwell, "Implementing co-investment and network sharing." *CERRE Centre on Regulation in Europe*. 2020. [Online] Available: https://cerre.eu/publications/telecom-co-investment-network-sharing-study/

[14] G. Földes, "Overview on corporate finance and regulatory framework of mobile network sharing in Europe. *Polgári szemle*, Vol. 18., 1–3. issue, pp. 309–323, 2022. DOI: 10.24307/psz.2022.1122, https://polgariszemle.hu/images/content/pdf/PSZ_2022_1_22.pdf

[15] NMHH, "Mobile market review 2021 H1." NMHH, the Hungarian NRA, 2022. [Online] Available: https://nmhh.hu/cikk/225890/Mobilpiaci_jelentes__2021_I_felev

[16] Ericsson, "Ericsson Mobility report." Ericsson, 2022. [Online] Available: https://www.ericsson.com/49d3a0/assets/local/reports-papers/mobility-report/documents/2022/ericsson-mobility-report-june-2022.pdf

[17] BEREC, "Report on infrastructure sharing." BEREC, 2018. [Online] Available: https://berec.europa.eu/eng/document_register/subject_matter/berec/reports/8164-berec-report-on-infrastructure-sharing

[18] GVH, "The GVH held unannounced inspections at the premises of Telekom and Telenor." GVH, the Hungarian National Competition Authority. 2018. [Online] Available: https://www.gvh.hu/en/press_room/press_releases/press_releases_2018/the_gvh_held_unannounced_inspections_at_the_premis.html

[19] DESI,"DigitalEconomyandSocietyIndex2019." 2019. [Online] Available: https://ec.europa.eu/commission/presscorner/detail/en/IP_19_2930

[20] DESI, "Digital Economy and Society Index 2020.", 2020. [Online] Available: https://digital-strategy.ec.europa.eu/en/library/digital-economy-and-society-index-desi-2020

[21] Opensignal, "The state of mobile network experience. Benchmarking mobile on the eve of the 5G revolution." 2019. [Online] Available: https://cdn.opensignal.com/public/data/reports/global/data-2019-05/the_state_of_mobile_experience_may_2019_0.pdf

[22] Analysys Mason, "Opex efficiency strategies for operators." Written by Venturelli, M., - Gabriel, C., - Brown, E., 2021. [Online] Available: https://www.analysysmason.com/research/content/reports/operator-opex-efficiency-strategies-rdns0/

[23] M. Bourreau, T. Shortall, W. Maxwell, "Cooperation between firms to deploy very high capacity networks." CERRE Centre on Regulation in Europe. 2020. [Online] Available: https://cerre.eu/wp-content/uploads/2020/11/REPORT_Cooperation-between-firms-to-deploy-very-high-capacity-networks_CERRE_November2020.pdf

[24] Empirica, "Mobile and Fixed Broadband Prices in Europe 2020 report." Empirica consultancy 11 prepared for the EC, 2021. [Online] Available: https://digital-strategy.ec.europa.eu/en/library/mobile-and-fixed-broadband-prices-europe-2020

[25] Telenor, "Telenor Hello Data campaign." Telenor, 2017, [Online] Available: https://www.youtube.com/watch?v=ihO6iv7XK88

[26] GSMA, "5G-era Mobile Network Cost Evolution." GSMA, 2019. https://www.gsma.com/futurenetworks/wiki/5g-era-mobile-network-cost-evolution/

[27] eNET, "National 5G strategy update. eNET consultancy led consortium for Hungarian Ministry of Innovation and Technology. by G. Kis, J. Tremmel, G. Földes, 2021, Non-public material.

[28] DESI, "Digital Economy and Society Index (DESI) 2021." Desi, 2021, [Online] Available: https://digital-strategy.ec.europa.eu/en/library/digital-economy-and-society-index-desi-2021

**Gábor Földes** Telecommunication economics expert with 20 years finance and regulatory experience in both academic and business controlling fields.

Graduated in 2004 at Budapest University of Economic Sciences, majored in finance. Also dealt with telecommunication as teaching assistant, participating in department researches, publications and preparation of academic book. After University, spent 15 years in telecommunication business at Deutsche Telekom, Magyar Telekom and Telenor Hungary in core controlling and granular performance management (Activity-Based Costing). Later worked for Hungarian telco National Regulatory Authority (NMHH) and support BEREC.

Currently works at Vodafone Intelligent Solutions Europe, VoIS Hungary in international controlling field.

In 2020 started a PhD techno-economic studies at Budapest University of Technology in cost efficient 5G rollout topic, covering mobile network sharing, TowerCo business model, virtual and open RAN multivendor concept, as well as hyperscaler co-investment. PhD was put on hold in 2022. Speaking on ITS Europe 2022 leading telco economics academic conference in Gothenburg on Chalmers University about mobile network sharing in CEE.

# 2023 IEEE International Mediterranean Conference on Communications and Networking

4 -7 September 2023 // Dubrovnik, Croatia

IEEE MeditCom

## CALL FOR PAPERS

*IEEE MeditCom is the conference of the IEEE Communications Society serving the Mediterranean area and surrounding countries. It gathers visionary researchers in academia and industry from all over the world to the shores of the Mediterranean Sea. IEEE MeditCom features a comprehensive and timely technical program, addressing the hottest topics in the areas of communications and networking. IEEE MeditCom 2023 solicits researchers in industry and academia to submit papers on a wide range of research subjects, encompassing theoretical and applied research. Original technical papers are sought, but are not limited, to the following areas:*

- 5G/6G Systems and Networks
- Antennas, Propagation, and Channel Modeling
- Big Data and Machine Learning for Communications
- Cloud Communications and Data-Center Networks
- Coding/Decoding Theory and Techniques for Communications
- Cognitive Radio and Dynamic Spectrum Access
- Communication and Information Theory
- Edge Computing, Edge Intelligence, and Fog Networks
- Energy Efficient Communications and Computing
- Image, Speech, and Signal Processing for Communications
- Integrated Sensing and Communications
- Internet of Things
- Massive MIMO and Cell-Free Massive MIMO
- Millimeter-Wave, Sub-Terahertz, and Terahertz Communications
- Molecular and Nanoscale Communications
- Network Applications, Services, and Management
- Network Architecture, SDN, NFV
- Next-Generation Multiple Access Schemes
- Next-Generation Physical, Link, and Network Layers Techniques
- Optical Communications and Networks
- Performance Evaluation, Simulation, Testbeds and Prototypes
- QoE/QoS Support and Cross-Layer Design
- Quantum Communications and Computing
- Reconfigurable Intelligent Surfaces and Holographic Surfaces
- Satellite and Space Communications
- Security, Privacy, Trust and Blockchain
- Semantic and Goal-Oriented Communications
- Smart Grids and Energy Networks
- Underground and Underwater Communications
- Vehicular Networks

## IMPORTANT DATES

| | |
|---|---|
| **Submissions Deadline:** | 1 May 2023 |
| **Acceptance Notification:** | 12 July 2023 |
| **Camera-Ready Submission:** | 22 July 2023 |

IEEE ComSoc

## Organizing Committee

**General Co-Chairs**
**Adriana Lipovac,** Univ. of Dubrovnik, Croatia
Hikmet Sari, NJUPT, China

**TPC Co-Chairs**
**Marco Di Renzo**, CNRS & CentraleSupelec, Paris-Saclay Univ., France
**Ana Garcia Armada,** Univ. Carlos III of Madrid, Spain

**Workshops Co-Chairs**
**Huseyin Arslan,** Medipol University, Turkey
**Guan Gui,** NJUPT, China

**Tutorials Co-Chairs**
**George Alexandropoulos,** NKUA, Greece
**Davide Dardari,** University of Bologna, Italy

**Panels Co-Chairs**
**Thierry Lestable,** TII, UAE
**Yi Wang,** Huawei, China

**Keynotes Chair**
**Stefano Bregni,** Politecnico di Milano, Italy

**Finance Chair**
**Anamaria Bjelopera,** Univ. of Dubrovnik, Croatia

**Local Arrangements Chair**
**Ante Mihaljević,** Univ. of Dubrovnik, Croatia

**Publication Co-Chairs**
**Mutlu Koca,** Bogazici University, Turkey

**Web & Social Media Chair**
**Ejder Bastug,** Nokia Bell Labs, France

**Publicity Co-Chairs**
**Ranga Rao V. Prasad,** TU Delft, Netherlands
**Eirini Eleni Tsiropoulou,** Univ. of New Mexico, USA
**Tomoaki Ohtsuki,** Keio University, Japan
**Marwa Chafii**, NYU Abu Dhabi, UAE
**Nury Gabriela Ramirez Cely**, HCL Technologies, Mexico

**www.ieee-meditcom.org**

# IEEE GLOBAL COMMUNICATIONS CONFERENCE

4-8 December 2023 // Kuala Lumpur, Malaysia

## CALL FOR PAPERS AND PROPOSALS

*Intelligent Communications for Shared Prosperity*

The 2023 IEEE Global Communications Conference (GLOBECOM) will be held in the warm and thriving city of Kuala Lumpur, MALAYSIA, from 4 to 8 December 2023. Themed "Intelligent Communications for Shared Prosperity," this flagship conference of the IEEE Communications Society will feature a comprehensive high-quality technical program including 13 symposia and a variety of tutorials and workshops. IEEE GLOBECOM 2023 will also feature an attractive industry program aimed at practitioners, with keynotes and panels from prominent research, industry and government leaders, business and industry panels, as well as a technological exhibit.

## TECHNICAL SYMPOSIA

- Cognitive Radio & AI-Enabled Networks
- Communication & Information Systems Security
- Communication QoS, Reliability & Modeling
- Communication Software & Multimedia
- Communication Theory
- Green Communication Systems & Networks
- IoT and Sensor Networks
- Mobile & Wireless Networks
- Next-Generation Networking & Internet
- Optical Networks & Systems
- Signal Processing for Communications
- Wireless Communications
- Select Areas in Communications
  - *Aerial Communications*
  - *Backhaul & Fronthaul: Communications, Networking & Signal Processing*
  - *Big Data*
  - *Cloud/Edge Computing, Networking, & Data Storage*
  - *E-Health*
  - *Integrated Sensing & Communication*
  - *Machine Learning for Communications*
  - *Molecular, Biological and Multi-Scale Communications*
  - *Quantum Communications & Computing*
  - *Reconfigurable Intelligent Surfaces*
  - *Satellite & Space Communications*
  - *Smart Grid Communications*
  - *Social Networks*
  - *Terahertz Communications*

### INDUSTRY FORUMS AND EXHIBITION PROGRAM

Proposals are sought for forums, panels, demos, seminars, and presentations specifically related to issues facing the broader communications and networking industries.

### TUTORIALS

Proposals are invited for half- or full-day tutorials in all communications and networking topics.

### WORKSHOPS

Proposals are invited for half- or full-day workshops in all communications and networking topics.

IEEE GLOBECOM® IEEE ComSoc | IEEE

## IMPORTANT DATES

**SYMPOSIUM PAPER SUBMISSION**
15 April 2023

**ACCEPTANCE NOTIFICATION**
1 August 2023

**CAMERA-READY SUBMISSION**
1 September 2023

**TUTORIAL PROPOSALS**
15 April 2023

**WORKSHOP PROPOSALS**
15 March 2023

**PANEL PROPOSALS**
1 June 2023

*Full details of submission procedures are available at globecom2023.ieee-globecom.org*

## ORGANIZING COMMITTEE

**General Chair**
Khaled B. Letaief (HKUST, Hong Kong)

**Executive Co-Chairs**
Borhanuddin Mohd Ali (UPM, Malaysia)
Hikmet Sari (NJUPT, China)

**Technical Program Co-Chairs**
Stefano Bregni (Politecnico di Milano, Italy)
Rose Qingyang Hu (Utah State University, USA)
Jiang (Linda) Xie (University of North Carolina at Charlotte, USA)

**Tutorial Program Co-Chairs**
Octavia Dobre (Memorial University, Canada)
Rolland Vida (Budapest University of Technology and Economics, Hungary)

**Workshop Program Co-Chairs**
Yacine Ghamri-Doudane (University of La Rochelle, France)
Guanyi Liu (China Mobile, China)
Christos Verikoukis (University of Patras, Greece)

**Industrial Forum & Exhibition Co-Chairs**
Luis M. Correia (IST – University Lisbon, Portugal)
Nordin Ramli (MIMOS Berhad, Malaysia)

**Patronage Co-Chairs**
Matti Latva-aho (University Oulu, Finland)
Azwan Mahmud (Multimedia University, Malaysia)

**Awards Chair**
George Rouskas (North Carolina State University, USA)

**Senior Advisor**
Hod Parman (TS Global Network Sdn. Bhd, Malaysia)

**GITC Advisor**
Hossam Hassanein (Queen's University, Kingston, Canada)

**GIMS Advisor**
Dong In Kim (Sungkyunkwan University, South Korea)

**globecom2023.ieee-globecom.org**

# Guidelines for our Authors

## Format of the manuscripts

Original manuscripts and final versions of papers should be submitted in IEEE format according to the formatting instructions available on

*https://journals.ieeeauthorcenter.ieee.org/*
*Then click: "IEEE Author Tools for Journals"*
*- "Article Templates"*
*- "Templates for Transactions".*

## Length of the manuscripts

The length of papers in the aforementioned format should be 6-8 journal pages.
Wherever appropriate, include 1-2 figures or tables per journal page.

## Paper structure

Papers should follow the standard structure, consisting of *Introduction* (the part of paper numbered by "1"), and *Conclusion* (the last numbered part) and several *Sections* in between.
The Introduction should introduce the topic, tell why the subject of the paper is important, summarize the state of the art with references to existing works and underline the main innovative results of the paper. The Introduction should conclude with outlining the structure of the paper.

## Accompanying parts

Papers should be accompanied by an *Abstract* and a few *Index Terms (Keywords)*. For the final version of accepted papers, please send the short cvs and *photos* of the authors as well.

## Authors

In the title of the paper, authors are listed in the order given in the submitted manuscript. Their full affiliations and e-mail addresses will be given in a footnote on the first page as shown in the template. No degrees or other titles of the authors are given. Memberships of IEEE, HTE and other professional societies will be indicated so please supply this information. When submitting the manuscript, one of the authors should be indicated as corresponding author providing his/her postal address, fax number and telephone number for eventual correspondence and communication with the Editorial Board.

## References

References should be listed at the end of the paper in the IEEE format, see below:

a) Last name of author or authors and first name or initials, or name of organization
b) Title of article in quotation marks
c) Title of periodical in full and set in italics
d) Volume, number, and, if available, part
e) First and last pages of article
f) Date of issue
g) Document Object Identifier (DOI)

*[11] Boggs, S.A. and Fujimoto, N., "Techniques and instrumentation for measurement of transients in gas-insulated switchgear," IEEE Transactions on Electrical Installation, vol. ET-19, no. 2, pp.87–92, April 1984. DOI: 10.1109/TEI.1984.298778*

Format of a book reference:

*[26] Peck, R.B., Hanson, W.E., and Thornburn, T.H., Foundation Engineering, 2nd ed. New York: McGraw-Hill, 1972, pp.230–292.*

All references should be referred by the corresponding numbers in the text.

## Figures

Figures should be black-and-white, clear, and drawn by the authors. Do not use figures or pictures downloaded from the Internet. Figures and pictures should be submitted also as separate files. Captions are obligatory. Within the text, references should be made by figure numbers, e.g. "see Fig. 2."
When using figures from other printed materials, exact references and note on copyright should be included. Obtaining the copyright is the responsibility of authors.

## Contact address

Authors are requested to submit their papers electronically via the following portal address:

https://www.ojs.hte.hu/infocommunications_journal/about/submissions

If you have any question about the journal or the submission process, please do not hesitate to contact us via e-mail:

Editor-in-Chief: Pál Varga – pvarga@tmit.bme.hu

Associate Editor-in-Chief:
Rolland Vida – vida@tmit.bme.hu
László Bacsárdi – bacsardi@hit.bme.hu

# Special Issue

of the **Infocommunication Journal**

Call for Papers

## *Advanced Distributed Optimization for Edge-Intelligence Communication Systems*

With the development of artificial intelligence and intelligent systems, multi-agent has been widely applied in practical edge-intelligence communication systems such as mobile edge computing networks, industrial Internet of Things (IoT), smart cities, smart home and smart grid networks. In these applications, multiple agents interact with each other by exchanging the training model or gradients, where a centralized node is often used to gather the status and reward of all the agents. This however brings a huge burden on the communication and a potential issue of information leakage and data island. To solve these issues, distributed optimization has been proposed to apply in the multi-agent learning, whereas federated learning is a typical form of distributed optimization to break dilemma of data island. There are still several fundamental challenges for the advanced distributed optimization in multi-agent learning for the edge-intelligence communication systems, from the aspects of convergence analysis, communication overhead and resource allocations, which have attracted much attention from both academy and industry.


This special issue collects the latest results emerging on the field of Cognitive Infocommunications.

*Special Issue Editors:*
**Prof. Lisheng Fan**
*Guangzhou University*
**George K. Karagiannidis**
*Aristotle University of Thessaloniki*
**Venki Balasubramanian**
*Federation University*

***Important dates:***

Submission paper deadline: **30th of September, 2023**
Notification first review: **30th of November, 2023**
Deadline for revised paper: **30th of December, 2023**
Camera Ready: **20th of February, 2024**

*Regarding manuscript submission information, please visit:*
**https://www.infocommunications.hu/for-our-authors**

# SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

## Who we are

Founded in 1949, the Scientific Association for Infocommunications (formerly known as Scientific Society for Telecommunications) is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals working in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

Besides its 1000 individual members, the Scientific Association for Infocommunications (in Hungarian: Híradástechnikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society.

## What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

## Contact information


President: **FERENC VÁGUJHELYI** • *elnok@hte.hu*

Secretary-General: **ISTVÁN MARADI** • *istvan.maradi@gmail.com*

Operations Director: **PÉTER NAGY** • *nagy.peter@hte.hu*

International Affairs: **ROLLAND VIDA, PhD** • *vida@tmit.bme.hu*

Address: H-1051 Budapest, Bajcsy-Zsilinszky str. 12, HUNGARY, Room: 502

Phone: +36 1 353 1027

E-mail: *info@hte.hu*, Web: *www.hte.hu*