# Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

August 2020 Volume XII Number 2 ISSN 2061-2079



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, National Council for Telecommunications and Information Technology (NHIT)

GÁBOR MAGYAR – president, Scientic Association for Infocommunications (HTE)

**Editorial Office** (Subscription and Advertisements)**:**
Scientic 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

# Special Issue on Quality Achievements at BME-VIK with Student Contributions in EFOP-3.6.2-16-013 – Guest Editorial

László Jereb

**The project EFOP-3.6.2-16-013, "*Thematic Research Collaborations for Innovative Informatics and Infocommunication Solutions*" (abbreviated as 3IN) started in September 2017. The abbreviation refers to the three participating institutions, Eötvös Loránd University (ELTE), Budapest University of Technology and Economics (BME), and Pázmány Péter Catholic University (PPKE), and to the three innovation areas in focus: Software Development and Information Security (A / Pillar), Infocommunication Networks and Cyberphysical Systems (B / Pillar) and Intelligent Data Analysis (C / Pillar).**

Over the past three years, a total of 150 Ph.D., MSc, and BSc students have been awarded scholarships at BME. Their research covered 15 topics in the three research pillars. Supervisors and mentors from six departments of the Faculty of Electrical Engineering and Informatics (BME-VIK) supported the work and scientific progress of the students. Their research has appeared in 80+ English papers in periodicals and conference proceedings, and 160+ presentations delivered at conferences and workshops organized abroad or in Hungary. The list of other publications contains about 300 TDK (Conference of Student Research Societies) reports and presentations, MSc and BSc theses, detailed research reports, and short summaries published in three special editions of the project summary booklet series.

The fundamental objective of the project was to support regional development in Hungary. Accordingly, the Central Transdanubia target region plays a unique role in the activities of the three universities. The series of dedicated local workshops held in Balatonfüred (BME), Martonvásár (ELTE), and Esztergom (PPKE) highlights the priority and impact of the regional dissemination of the project results.

This special issue of the Infocommunications Journal offers a unique opportunity to the research students at BME since the nine representative papers chosen from the three pillars and 15 research topics present their results to the experts and the professional community. The articles reflect well the excellent cooperation between the students performing the research, and their respective supervisors and mentors guiding and helping their scientific work during the entire project. Many cases of the gradual transition from Ph.D. to mentor, MSc to Ph.D., and BSc to MSc prove the significant impact of the project.

The rich spectrum of the topics of the papers is representative of the broad coverage of research fields (and supporting departments) by the project. The first four papers cover infrastructure and security-related problems. Simon et al. propose a sidecar-based solution to evaluate available resources in a virtual environment for real-time monitoring with direct applicability to Virtualized Network Functions.

Marosits et al. introduce a quantum random number generator (QRNG) based on the phenomena of amplified spontaneous emission (ASE), describe the real-time generation hardware and software implementation. Their results are open for the broad public by a web page offering a real-time random bits generator.

Kobor et al. also deal with quantum communication. Their particular focus is on the physical layer of an optical system realizing quantum key distribution. They evaluated the weak points using simulation and suggested specific polarization-dependent optical devices to improve the transmission quality significantly.

In the last paper of this section, Ládi et al. propose a graph analysis based method (GrAMeFFSI) that can restore the message formats and field semantics of (potentially undocumented) binary protocols from network traces, and demonstrate the usability of the approach in the case of two standardized protocols, Modbus, and MQTT.

The next two articles take us into the world of data analysis, discussing methodological issues. Papp et al. investigate the known drawback of many unsupervised machine learning algorithms. Data clustering data based on similarity metrics often ignores other types of relations between the individual data. The paper presents conditions for the construction of a weighted graph used in spectral clustering, preserving the hierarchical structure of the dataset.

Pilinszki-Nagy et al. compare the Hierarchical Temporal Memory's (HTM) performance in terms of accuracy, speed, and memory complexity to the deep learning-based LSTM (Long Short-Term Memory) network.

The final three papers show inspiring examples of the use of the outcomes of the project results for very different application domains. Fábián et al. propose in the first paper an approach for creating synthetic, representative datasets consisting of embeddings and demographic data of several people, and show that even simple machine learning models are able to reach a proportion of successfully re-identified people between 6.04% and 28.90%, depending on the population size of the simulation.

Alekszejenkó et al. make decisions based on mathematical algorithms borrowed from information technology and adapt them to the traffic lights' optimal and fair timing in intelligent urban traffic control. The results show that the optimal scheduling based traffic light control can outperform the traditional light programs in extraordinary and especially rapidly evolving situations.

In the closing paper, Varnyú et al. aim at reducing the noise in positron emission tomography (PET) by comparing the most powerful image denoising filters, improving both image quality and execution time. The non-linear methods compared include the Gaussian, the bilateral, the guided, the anisotropic diffusion, and the non-local means filters, in static and dynamic PET reconstructions.

**László Jereb** graduated from the Budapest University of Technology in 1971, then received the Candidate of Science, and the Doctor of the Hungarian Academy of Science (MTA) titles, in 1984 and 2004, respectively. At BME, his main research interest included reliability analysis, multi-layer network planning, and performance modeling and evaluation of networks.

He launched the business information technology track in 2002 at the University of West Hungary. He served as the Dean of Faculty of Wood Industry Engineering between 2008 and 2013. He is currently professor emeritus of the Budapest University of Technology and Economics and the University of Sopron. Since 2014, he coordinates the BME participation focussed on innovation projects and innovation and entrepreneurship education in EIT Digital. Since 2017, he leads the BME activities in the project EFOP-3.6.2-16-2017-00013.

The architectural perspective of the BME Knowledge Center under construction in Balatonfüred with the support of EFOP 4.2.1-16-2017-00021.

# Our reviewers in 2019 and 2020

*The quality of a research journal depends largely on its reviewing process and, first of all, on the professional service of its reviewers. It is my pleasure to publish the list of our reviewers of 15 countries in 2019 and 2020 (so far) and would like to express my gratitude to them for their devoted work.*

*Your Editor-in-Chief*

**Gusztáv Adamis**
BME, Hungary
**László Bacsárdi**
University of Sopron, BME, Hungary
**Péter Baranyi**
Széchenyi István University, BME, Hungary
**Bostjan Batagelj**
University of Ljubljana, Slovenia
**Gergely Biczók**
BME, Hungary
**János Bitó**
BME, Hungary
**Daniela Cancila**
CEA, France
**Tamás Gábor Csapó**
BME, Hungary
**Tibor Csöndes**
Ericsson Research, Hungary
**László Csurgai-Horvath**
BME, Hungary
**Tadeusz P. Dobrowiecki**
BME, Hungary
**Andrea Farkasvölgyi**
BME, Hungary
**Hugo Ferreira**
INESC TEC, Portugal
**Péter Fiala**
BME, Hungary
**Attila Frankó**
AITIA International Inc., Hungary
**Csaba Gáspár**
DMLab, BME, Hungary
**Juraj Gazda**
Technical University of Kosice, Slovakia
**István Gódor**
Ericsson Research, Hungary
**László Göcs**
John von Neumann University, Hungary
**Bálint Gyires-Tóth**
BME, Hungary
**László Gyöngyösi**
BME, Hungary
**Gábor György Gulyás**
INRIA, France
**Csaba Hegedűs**
TATA Consulting, Hungary
**Attila Hilt**
Nokia Networks, Hungary
**Gergely Hollósi**
BME, Hungary
**Máté Horváth**
BME, Hungary
**Péter Horváth**
BME, Hungary
**Sándor Imre**
BME, Hungary
**Haris Isakovic**
Technical University of Vienna, Austria
**Stavros Iezekiel**
University of Cyprus, Cyprus
**Zoltán Jakó**
BME, Hungary
**Ferenc Nándor Janky**
Morgan Stanley, Hungary
**Tomaž Javornik**
Jožef Stefan Institute, Slovenia
**Zsolt Csaba Johanyák**
John von Neumann University, Hungary
**Zsolt Kollár**
BME, Hungary
**Andrey Koucheryavy**
St. Petersburg State Univ. of Telecomm's, Russia
**Stavros Koulouridis**
University of Patras, Greece
**Szilveszter Kovács**
University of Miskolc, Hungary
**Zsolt Krämer**
BME, Hungary
**Dragana Krstić**
University of Niš, Serbia
**János Ladvánszky**
Ericsson Research, Hungary
**Jan Latal**
VŠB-Technical University of Ostrava, Czech Rep.
**Tamás Lévai**
BME, Hungary
**Gábor Magyar**
BME, Hungary
**Vashek Matyáš**
Masaryk University, Czech Rep.
**Gyula Mikó**
BHE Bonn Hungary Electronics Ltd., Hungary
**Farshad Miramirkhani**
Isik University, Turkey
**István Moldován**
BME, Hungary
**Miklós Molnár**
LIRMM, France
**Ioannis Moscholios**
University of Peloponnese, Greece
**István Nagy**
DMLab, BME, Hungary
**Lajos Nagy**
BME, Hungary
**Gábor Németh**
BME, Hungary
**Peter Olasz**
Nokia Networks, Hungary
**Péter Orosz**
BME, Hungary
**László Osváth**
BME, Hungary
**Béla Pataki**
BME, Hungary
**András Pataricza**
BME, Hungary
**Balint Peceli**
Evopro Innovations Ltd., Hungary
**Sándor Plósz**
BME, Hungary
**Rama Rao T**
SRM Institute of Science and Technology, India
**Gábor Recski**
BME, Hungary
**Patrik Reizinger**
BME, Hungary
**Rafael Rocha**
Inst. Superior de Engenharia do Porto, Portugal
**Tamás Skopkó**
BME, Hungary
**Balázs Sonkoly**
BME, Hungary
**Gabor Soós**
Magyar Telekom, Hungary
**Gábor Szűcs**
BME, Hungary
**Péter Tatai**
AITIA International Inc., Hungary
**László Toka**
BME, Hungary
**Eszter Udvary**
BME, Hungary
**John Vardakas**
Iquadrat Informatica S.L., Spain
**József Varga**
Nokia Bell Labs, Hungary
**Pál Varga**
BME, Hungary
**Rolland Vida**
BME, Hungary
**Attila Zólomy**
BME, Hungary
**Zoltán Zsóka**
BME, Hungary

# Sidecar based resource estimation method for virtualized environments

Csaba Simon[1], Markosz Maliosz[2], Miklós Máté[3], Dávid Balla[4], and Kristóf Torma[5]

***Abstract*—The widespread use of virtualization technologies in telecommunication system resulted in series of benefits, as flexibility, agility and increased resource usage efficiency. Nevertheless, the use of Virtualized Network Functions (VNF) in virtualized modules (e.g., containers, virtual machines) also means that some legacy mechanisms that are crucial for a telco grade operation are no longer efficient. Specifically, the monitoring of the resource sets (e.g., CPU power, memory capacity) allocated to VNFs cannot rely anymore on the methods developed for earlier deployment scenarios. Even the recent monitoring solutions designed for cloud environments is rendered useless if the VNF vendor and the telco solution supplier has to deploy its product into a virtualized environment, since it does not have access to the host level monitoring tools. In this paper we propose a sidecar-based solution to evaluate the resources available for a virtualized process. We evaluated the accuracy of our proposal in a proof of concept deployment, using KVM, Docker and Kubernetes virtualization technologies, respectively. We show that our proposal can provide real monitoring data and discuss its applicability.**

***Index Terms*—Computer network management, Network function virtualization.**

## I. INTRODUCTION

MODERN, high performance telecommunication software is implemented as a collection of stateless microservices for maximum scalability and fault-tolerance. These microservices have so far been running in controlled environments with known performance characteristics. In the near future, however, these systems must be able to work in any environment, even in heterogeneous ones, and ones with volatile resource availability [1]. Moreover, in a virtualized environment the available resources reported by the system may not accurately reflect the amount of resources that are physically available. Therefore, if the telecommunication systems want to perform load balancing, autoscaling or overload prediction, these applications need to measure their own performance, report it to the framework to provide sufficient information to deduce the available resources.

Porting such measurement tasks onto stateless microservice applications is challenging, since new resource monitoring approach should be applied in order to circumvent the resource estimation ambiguity. In this paper we examined the feasibility of using a separate measurement application for the estimation of the available resources. This measurement application runs in a container or a virtual machine separate from the main telecommunication application. This configuration is called "sidecar" to reflect on the similarities with attaching a sidecar to a motorbike and is a well-known usage pattern in virtualized computing systems [2].

The main goal of this paper is to validate the feasibility of performance measurements from a sidecar. In this paper we focus on telecommunication (telco) applications that, compared to generic webservices, must fulfill much stricter Service Level Agreements, and they are much vulnerable to insufficient (or less than agreed) resource sets. Therefore a correct evaluation of the resources available for a given telco app is crucial to operate within the agreed parameters. In principle, increasing resource usage by the telco application results in degraded computing performance in the sidecar, but the sensitivity and the accuracy of this method were previously unknown. In order to eliminate the dependency on (potentially) bogus CPU usage resource reporting available from inside a virtualized space, we monitored the completion time of a reference task as the main indicator of the computing performance of the underlying infrastructure.

In the next Section we present the technologies used in the investigated virtualized environments, present a problem statement and a literature survey. In Section III we introduce our proposal and present a proof of concept deployment of our proposal, based on which we present a detailed measurement-based evaluation of it. In Section IV we discuss the possible limitations and the applicability of our proposal and finally we conclude our work.

## II. RELATED WORK

In this section we present the virtualization aspects of the infrastructure that are relevant to our work first. To the best of our knowledge, our approach described in this paper was not published before. Still, the wider topic of performance monitoring aspects of virtualized applications has been intensively investigated in the last decade and has a vast literature. In the related work part of this section we present the typical approaches to mitigate the performance monitoring problem of telecommunication systems deploying Virtualized Network Functions (VNFs). We also present a set of works that inspired us to use service completion times to characterize the resource set available to an application.

[1,2,3,4,5]The authors are with HSNLab, Dept. of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary. E-mail: {simon, maliosz, mate, balla, torma}@tmit.bme.hu.

[4,5] BME Balatonfüred Student Research Group, Hungary

DOI: 10.36244/ICJ.2020.2.1

### A. Virtualization technologies

Virtualization is a technology that introduces a layer of abstraction between computing, storage and networking hardware, and the applications running on it. Thus, the underlying physical resources (CPU, memory, disk and network) are shared, and there can be multiple systems (or virtual machines - VMs) running simultaneously and concurrently on the same host. There are several approaches to implement virtualization, but in modern cloud systems there are two alternatives that are used: the host-based and the operating system level virtualizations.

The Kernel-based Virtual Machine (KVM) is a hypervisor module of the Linux kernel [3]. It allows running guest operating systems in a virtualized environment. The KVM kernel module is only a hypervisor, the virtual devices, networking etc. must be supplied to the VM by the virtualization program, and the most widely known one is QEMU [4]. QEMU implements CPU emulation in software, but its qemu-kvm extension uses KVM instead of its soft-cpu implementation. Finally, we may use *libvirt* library [5] manage VMs, including *cgroup* policy groups for resource policy control [6]. Since *cgroups* is a powerful and important mechanism used by us also for both VM and container resource control, we describe it in detail in the following section.

The operating system level virtualization, also known as containerization, does not virtualize the host hardware as other types do. Instead, it virtualizes the kernel of the host. Opposed to the host-based virtualization, the containers do not need a hypervisor, instead they run directly within the host machine's kernel. The isolation and resource control tasks are assured by the namespaces [7] and control group (*cgroup*) [6] mechanisms of the kernel, respectively. The most well-known container technology is Docker [8]. An important technology within the container ecosystem is Kubernetes [9], a container management framework. Kubernetes extends the process-oriented approach of Docker and focuses on services instead. In Kubernetes, the service is implemented by a set of connected containers, called pods. In Kubernetes, the pods are the basic unit of scaling, and per-pod resource usage pattern can be specified.

The resource usage of a Linux system by default is governed by *cgroups*. The CPU scheduler of Linux shares the CPU time among the process groups according to their *cpu.shares* value; the default value is 1024. E.g., if there is one CPU, and two groups want to use it fully, by default they both get 50% share of the CPU. If we change the shares of one group to 512, that group will receive 33%, and the other will receive 66%. This division happens hierarchically: the sub-groups receive the CPU percentage of their parent group. When Docker is active on the host, it inserts its own slice, named *docker*. Similarly, QEMU based VMs get their own top-level slice, called machine-slice. As a consequence, Docker containers and KVM/QEMU VMs are handled in isolated resource buckets (cgroup slices) by the host-level cgroups scheduler. Kubernetes has its own mechanism that configures the resource reservation quotas of the containers started in its pods [10]. In our paper we use the so called burstable mechanism, where each container specifies its resource usage intent (*request*) but lets the Kubernetes framework to scale the resources according to the total available set. Then Kubernetes makes sure that the allocated resources to different containers keep the ratio of the declared *requests*.

### B. Related work

The authors publishing in this field mostly focused their efforts on providing a working solution to address the monitoring needs of the cloud native telecom systems that emerged since the beginning of the 2010s. As part of these efforts, several solutions were proposed to provide accurate resource usage in cloud native telecom systems. Paper [1] introduces a complete monitoring framework for cloud native 5G systems. Still, it considers that the access to the physical node metrics is granted.

A more academic approach is followed in [11], where realtime prediction and long term forecasting is used to support the autoscaling process for container-based telecom microservices. The authors exploit the specific nature of typical telecom services due to the repetitive nature of human behavior. Still, this approach relies on generic Kubernetes monitoring technologies and the author's custom monitoring container, if they have access to the real performance data from the underlying host system.

Several works analyze the statistical characteristics of the observed resource usage parameters for the VNFs and infer the availability and sufficiency of the resources in the system based on these. A good example of these works is [12], where the skewness of the probability distribution of per VNF CPU usage is used as an indicator of system-wide resource availability. The authors show that their proposal can be used to provide automatic notifications in case of system overload. Nevertheless, this approach also requires the access of host level information or Docker API at the host.

The above cited articles [1][11][12] are representative for the prior work in this field. Due to lack of space we do not offer further insight into other proposals, but the interested reader is referred to the related work sections of these papers in order to get a wider knowledge of the state of art in this area. Our solution will differ from these, since our novel approach avoids any use of any information that may be obtained from the host.

As already described above, the resource monitoring approaches observed several parameters when tried to model the available resource sets, not only the CPU usage. This gave us the idea to verify if exists such a parameter that can be measured from inside the virtualized space and is a good indicator of the available resources (e.g., CPU power). Based on our literature survey we have seen that the service completion times, the resources consumption (i.e., the allocated resources, if the service uses all available resources) and the user demands are strongly dependent on each other. We found that relevant works were published since the mid-2000s and mostly relate to the field of BigData. A good

introduction of this approach is found in [13], where the authors measured both the response times of classical industrial IT applications and the CPU utilization, and used it to estimate the volumes of user demands. The approach of measuring the service completion time later was used in paper [14] to offer an accurate scheduling mechanism, where based on demand (i.e., job size) and resource availability (number of parallel worker instances) a certain completion time can be guaranteed.

In our scenarios user demand can be easily obtained, either by the framework itself or by the application by monitoring the incoming request rate. The service completion time can be measured from inside the virtualized space. Thus, based on [13][14] we supposed that observing these two parameters, we can provide a good estimation of the compute resources, and we proposed a method, which is introduced in the next section.

## III. SIDECAR BASED RESOURCE ESTIMATION METHOD AND PROOF OF CONCEPT

### A. Sidecar based resource estimation

As described in the previous section, we propose to evaluate the resource usage of a virtualized function (or application) by observing the duration of an application. In practice there is a large variety of VNFs in a telecom system, and each of these VNFs have their own resource usage characteristics, which also depend on the current load. Therefore, the measurement of the VNF is not useful for this role. Before using the measured response time of a VNF to evaluate the resources it used during the observation period, a detailed profiling of the VNF would be needed. Even if this is doable, as VNF vendors may be required to do this profiling before shipping their product, the management of release schedule and continuous update of this data in a large telecommunication system is not practical.

As an alternative we propose to use the *same* application for *every* VNF and use *this* application as a benchmark. This application should be selected such as it correlates with the resource set allocated to it and it has a stable performance.

We propose to deploy this monitoring application as a sidecar together with all the VNFs that require resource estimation. This sidecar should run in the same virtualized environment, as the "target" VNF. In the case of VMs or Docker containers both the monitoring sidecar application and the target VNF should run on the same machine, with further conditions detailed in Section IV. In the case of Kubernetes based deployment, the monitoring sidecar application and the target VNF should be deployed within the same pod.

### B. Load emulation

In our work we used the *stress-ng* utility [15] to generate load on the CPU. It is a flexible utility capable of running several different stressor routines in any number of parallel processes. Therefore, we considered to be versatile enough to model a generic VNF during our evaluations. It was not designed to be a benchmark, but we judged that its metrics (called bogo operations/sec, referred to as *bogo ops*) are sufficiently accurate for our purposes. Thus, we used the same tool for both generating load (*gen*) and serving as a monitoring probe (*mon*).

We mainly used the *cpu* stressor, which contains more than 70 different stressor algorithms, and the default setting is to loop over all of them repeatedly. These algorithms perform different numeric computations, and together they stress of the various arithmetic units of the CPU. Nevertheless, we also tested the memory stressor, and two stressors using system-calls (executing timer calls and pipe operations).

Stress-ng can print the number of iterations it ran within the specified time limit with the option *--metrics-brief*. It cannot report per-process results, just the total for all the stressor processes of the same type. For continuous monitoring of the performance of stress-ng it must be run in an endless loop with short timeout of 20 s. This reporting period is much longer than the measurement periods typical for monitoring systems in production (1 s), but in our evaluation let stress-ng perform several hundred iterations in all scenarios to minimize quantization errors. In a real-life scenario, running VNFs under heavy load, a 1 s measurement period would lead to similar accuracy. The overhead of restarting stress-ng is negligible.

Based on extensive tests we decided to configure four stress-ng stressors during the tests. For both the *gen* and *mon* roles, we run the following operations to generate their load:

- CPU – integer and floating-point mathematical operations run in user mode
- Memory – *mmap()/munmap()* calls with 256 MB data
- Timer – sets one million timers each second, and counts how many of them are completed successfully
- Pipe – moving data through Linux pipes. The size of the pipe is 512 MB, and the data size is 4 KB (equals the memory page size).

The detailed parameter setup is shown in Table I. It can be seen that the parameters, and implicitly the load of the *mon* process is independent of the monitored *gen* process. Thus the cost of our solution is constant. In a real life deployment scenario the load level can be adjusted to the available resources.

TABLE I
*THE PARAMETERS OF THE STRESS TOOL USED TO TEST THE SIDECAR SCENARIO*

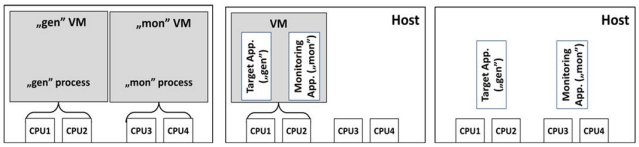
| Stressor type | "gen" process | "mon" process |
|---|---|---|
| CPU | --cpu 1 | --cpu 1 |
| Virtual Memory | --vm 1 | --vm 1 –vm-bytes 20 |
| Timer | --timer 1 | --timer 1 |
| Pipe | --pipe 1 | --pipe 1 |

### C. Configuration of the virtual environments

During our measurements we used both KVM/QEMU VMs and Docker containers. The VMs used in our tests were provisioned with Vagrant, and depending on the scenario, we run a single VM or two VMs. When two VMs were provisioned, one VM acted as the target application, generating the load to be monitored (*gen*). The other VM acted as the monitoring VM (*mon*). We allocated two CPU cores

and 1 GB RAM for each. When a single VM was used (e.g., in Section V.A), only one of the VMs was started. When the stress-ng process was containerized (e.g., in Sections IV.A and IV.B), we used our custom Docker image, created from Ubuntu 18.04.1 LTS, and installed a stress-ng v.0.09.25. The Docker container was run with no resource limits.

Depending on the measurement setup, we had four arrangements. In the first one we had two VMs and in each VM we run a stress-ng process, as shown in Fig. 1 a) and the measurements on this setup are discussed in Section IV.A.




(a) (b) (c)

Fig. 1. Sidecar scenarios with a) two VMs, b) two containers run in single VMs, and c) with two containers run on the host, respectively.

Note that the pinning of VMs might differ from the one illustrated in Fig. 1 a), according to the details given in Section IV.A. The parameters of these two stress-ng processes were the ones already shown in Table I.

A second measurement setup used only one VM, both the *gen* and *mon* processes were containerized, and these two containers were run within the VM. This setup is shown in Fig. 1 b) and is discussed in Section IV.B. A third measurement setup without VMs used only Docker containers, where the *gen* and *mon* containers were run on the host. The containers shared all the resources of the hosts and this setup is illustrated in Fig. 1 c) and is discussed in Section IV.B. Finally, we had a fourth measurement setup, where two containers were run in a single pod. The measurements with this setup are discussed in Section IV.C.

We run our test on desktop PCs, the detailed hardware specification is shown in Table II.

TABLE II
*THE HARDWARE USED DURING TO EVALUATE THE SCENARIOS*

| Name | CPU type | Frequency [GHz] | RAM [GBytes] |
|---|---|---|---|
| PC1 | Intel Core i5-2400 | 3,1 | 8 (DDR3) |
| PC2 | Intel Core2 Quad Q6600 | 2,4 | 6 (DDR2) |
| PC3 | AMD Athlon 64 X2 5050e | 2,6 | 6 (DDR2) |
| PC4 | Core i5-3320M | 2,6 | 8 (DDR3) |

## IV. EVALUATION OF THE PROPOSAL

In this section we run three set of experiments to evaluate our proposal from III.A in the test environment described in the previous section.

### A. VM based deployments

The first sets of experiments were conducted with VM based deployments. The measurement setup is illustrated in Fig. 1 a), where machine *mon* is the sidecar VM that monitors its own performance, and tries to deduce the resources used by the *gen* process from the other VM, based on its own performance.

We limited the CPU usage of *stress-ng* with *cgroups* policies applied to the processes representing QEMU's virtual CPUs on the host. We used the *cpuset cgroup* to pin the vCPUs to specific physical CPUs, and the *cpu cgroup*'s *cfs_quota_ms* parameter to impose a quota on per-VM level. Each presented measurement point is the aggregation of 10 experiments.

When each VM only have 1 vCPU allocated, it can be the same cores for both VMs, or different. Fig. 2 shows the performance of *mon* when it shares a single CPU with *gen*. The different colors correspond to different loads on *gen*. When there is light load on *gen*, the measured performance of *mon* VM correlates with the load of *gen*. But when *gen* is at least 50% loaded, the performance of *mon* is independent of the load, this setup is thus not suitable for detecting overload on the telecom application.

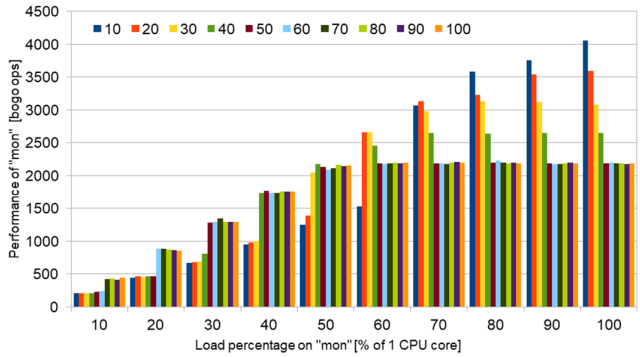



Fig. 2. Performance measured in bogo ops, when "gen" and "mon" share a single CPU. The colored bars correspond to different loads on "gen", expressed as % of 1 CPU core capacity.

When each VM only have 1 vCPU allocated, but they are mapped the different physical CPU cores, the performance figure differs from the previous case, as shown in Fig. 3. In this case when *gen* is getting close to the maximum load, the performance of *mon* gets a noticeable bump. Note however, that this bump starts at around 70% percent load on *gen*, which is still quite far from its maximum capacity. Another problem with this setup is that we are loading only 1+1 cores of a 4-core CPU; thus, the performance bump of *mon* comes from the raised CPU frequencies under heavy load. In a real deployment the applications usually try to put load on all available CPU cores, resulting in different performance profiles.

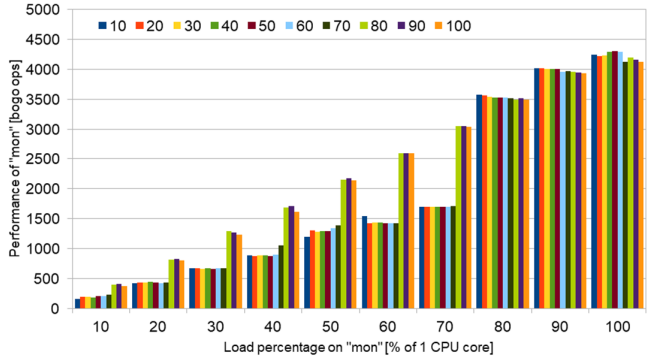



Fig. 3. Performance measured in bogo ops, when "gen" and "mon" run on different CPU cores. The colored bars correspond to different loads on "gen", expressed as % of 1 CPU core capacity.

When in our 4 core CPU host machines two vCPUs are allocated to both VMs, the CPU cores assigned to the VMs can be all different, only one shared, or both shared between the two VMs. The figures for the "all different" and the "all shared" CPU core scenarios look identical to the results shown in Fig. 2 and Fig. 3, respectively. This was the expected behavior and we do not show the results. Nevertheless, we observed a different behavior in the case when the VMs share one core, but they both have one independent core, as well. Fig. 4 shows that this scenario is quite like to the single shared CPU core scenario (i.e., Fig 2), but it inherits the sensitivity threshold of the single different CPU core scenario. The load percentages on the figure are doubled in this case, because maximum load for 2 CPUs is 200%.

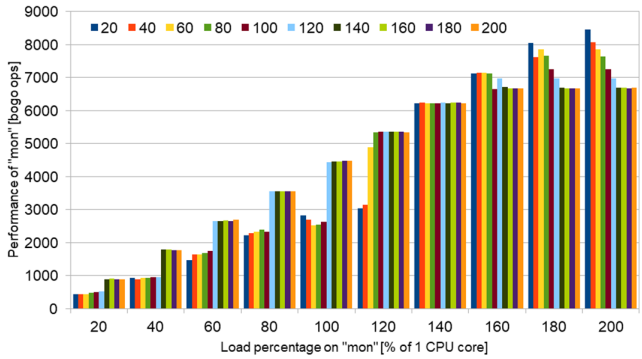



Fig. 4. Performance measured in bogo ops, when "gen" and "mon" share one of their CPU cores. The colored bars correspond to different loads on "gen", expressed as % of 1 CPU core capacity.

We also created a scenario, where *gen* had access to all four CPU cores, and *mon* had only one vCPU. Probably this scenario models the best a real deployment of a telco application getting the most computation resources possible, with a sidecar VM with limited CPU usage measuring it. Fig. 5 shows the results for this scenario (note that the maximum load of 400% corresponds to full utilization of 4 CPU cores). It is largely identical to the previous results: *mon* can detect changes in the load of *gen*, when that is low, however, when the load of *gen* is high, *mon* becomes blind.

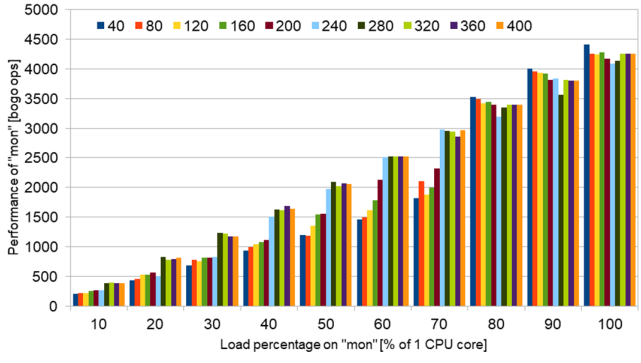



Fig. 5. Performance measured in bogo ops, when "gen" and "mon" share a single CPU core. The colored bars correspond to different loads on "gen", expressed as % of 1 CPU core capacity.

Note that in all the above scenarios *mon* can perform its load detection while generating small load itself. This is a nice property, as it allows running the performance monitoring sidecar with low impact on the telco application.

### B. *Docker container-based deployments*

In this section we describe our results on testing the sidecar scenario when the processes were containerized. Similarly to the previous section, the container emulating the load of the target application was named *gen*, and the monitoring container was named *mon*.

In the case of container-based deployments we did not experience the dependence of the accuracy of load detection on the load level of the *mon* or the *gen* processes, as seen in the VM based deployments. Therefore in this section we compare the outcome of experiments with the same loads, but run on computers with different resource sets.

We compared two use cases: in the first case the containers run on the host (see Fig. 1 c), corresponding to a bare metal deployment of Docker containers. In the second one the two containers were run within a KVM/QEMU VM (see Fig. 1 b), modelling the widely used practice of deploying a container in a VM of a datacenter. The details of the VM, container setup, and the parameters of the load generator are all described in section III.

In these measurements the *stress-ng* was started at once (with the 4 stressors of different types set as shown in Table I), but we present them in four different charts: Fig. 6 for the CPU stressor, Fig. 7 for the memory stressor, Fig. 8 for the timer stressor and Fig. 9 for the pipe stressor.

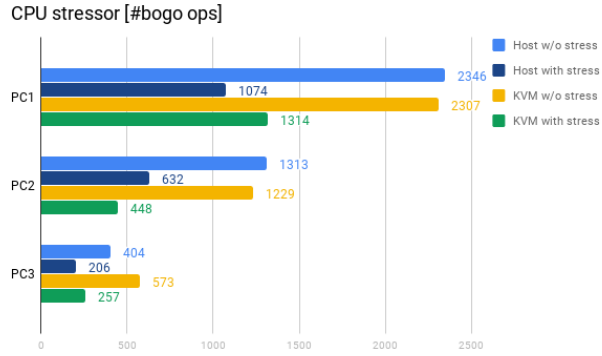



Fig. 6. Container-based scenario results with the CPU stressor.

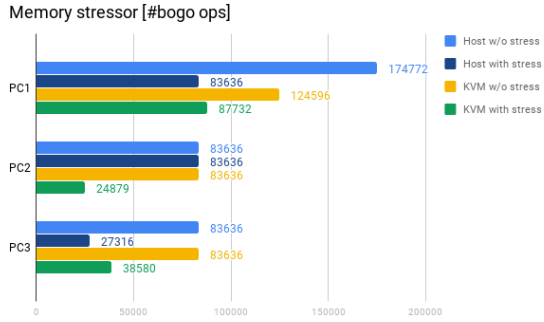



Fig. 7. Container-based scenario results with the memory stressor.

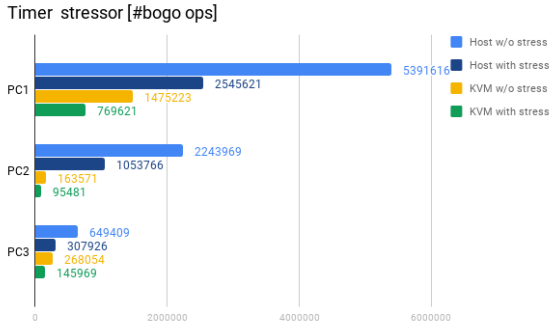



Fig. 8. Container-based scenario results with the timer stressor.

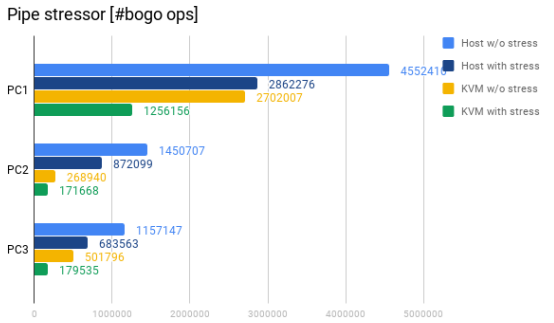



Fig. 9. Container-based scenario results with the pipe stressor.

In all four cases, for both host and VM based measurements it is clearly observable the effect of the stress on the *gen* container. It also can be seen that VM-based measurements result in lower values. However, the difference between the host-based and VM-based values depends on the stressor types: for the memory stressor the differences may be minimal (depends on the motherboard architecture and RAM type, not only on CPU type), whereas for the timer stressor we observed extreme differences.

For the stressors triggering *timer()* and *pipe()* system calls are much more sensitive to the computer architectures and react much more in terms of absolute value to the presence of load. Whereas this is useful to detect differences in both load and computational power, it has the drawback that it is volatile and has larger variance compared to the *cpu* and *memory* stressors.

In Table III we summarized the relative differences among the three PCs, calculated based on the *bogo ops*, as reported by the CPU stressor of *mon*.

TABLE III *CPU PERFORMANCE COMPARISON (RELATIVE TO PC1)*

| Name | Measured by "mon" container | CPUboss.com benchmark values |
|---|---|---|
| PC1 | 1 | 1 |
| PC2 | 0,56 | 0,46 |
| PC3 | 0,17 | 0,24 |

In a separate column we show the *cpu score* based relative performance of the 3 CPUs, as provided by the *cpuboss.com* independent CPU benchmark site. It can be seen that our measurement accurately profile the 3 computers (note that the motherboard and RAM configurations correspond to the performance levels of the CPUs, thus this did not introduce further bias in the measurements).

### C. Kubernetes based deployments

In the third experiment series we tested the sidecar scenario in a Kubernetes cluster. We deployed a pod running the two containers (*gen* and *mon*). Each container ran one stress-ng process each. The stressors were parameterized according to Table I, with the notable exception of starting 4 parallel CPU stressors in the *gen* container in order to allow it to consume as much CPU as it can.

During the tests, we started an external stress in a second pod, which stole resources from our pod. The *mon* container repeated the measurements in an infinite loop. The goal was to let the *mon* container measure the level of resource degradation.

The resource definition for the pod was set for CPU only. Within our pod, the *gen* container requested 1800 milli cores, and the *mon* container requested 200 milli cores of CPU, respectively. The external load that supposed to stole resources from our pod requested 1000 milli cores of CPU. The resource allocation policy was burstable (see Section II.A) and the pods were scheduled on PC2. The measurements have shown that the performances of the two containers (*mon* and *gen*) correlate. We verified the CPU usage on the host using the *top* tool. At the beginning of the experiment the pod generating the external load was not deployed, then we started the external load. The CPU consumption of the *gen* and *mon* containers before and after the external load is started is shown in Table IV. Initially the *gen* container uses as much resources as it can (3.8 CPUs). After the external load steals some resources (it gets 1.2 CPUs), the *gen* container can consume only ~60% of this resource (2.4 CPUs). The resource usage of the *mon* container scales down in a similar manner.

TABLE IV *THE CPU CONSUMPTION OF THE OBSERVED CONTAINERS DEPLOYED INTO A KUBERNETES CLUSTER, AS FUNCTION OF EXTERNAL LOAD*

| External load? | CPU consumption of the "gen" container [milli cores] | CPU consumption of the "mon" container [milli cores] | CPU consumption of the "mon" container [bogo ops] |
|---|---|---|---|
| NO | 3777 | 213 | 134 |
| YES | 2410 | 118 | 68 |

The 4th column of Table IV shows the measured values, as recorded by the "mon" container (expressed in *bogo ops*). The resource degradation level measured by the *mon* container is like the one observed at the host (3rd column) but is not exact match. This is because that the *stress-ng* load does not depend solely on the CPU usage. In practice this method must be calibrated to the proper application it is supposed to measure.

## V. DISCUSSION OF RESULTS

The measurement results presented in this study were done on computers with four cores, and the results shown in the

previous section suggest that sidecar containers can detect if the main container is loaded just by monitoring the CPU frequencies, even if the two are pinned to different CPU cores.

### A. The effects of the CPU frequency modifying mechanisms

The modern CPU architectures apply several optimization features, resulting in dynamic CPU resource availability that adapts to the load variations. Most of these features were introduced to increase the power consumption efficiency. The Intel CPUs implement frequency scaling in hardware, called *SpeedStep* technology. When a workload is deployed on one core, this technology raises the clock frequencies on all cores; the fewer cores are loaded, the higher their frequency can go.

Additionally to the above feature, a mechanism called *turbo frequency adjustment* aims to allow higher peak performances for short periods If multiple cores are loaded at the same time, their clock frequency drops below the maximum turbo frequency; thus, the overall computing capacity of the CPU doesn't scale linearly with the number of threads running.

We also ran some of the measurements detailed in section IV.A on a computing cluster, where the servers had CPU frequency scaling turned off in the BIOS. The measurement results confirmed that when the CPU frequencies are constant throughout the tests, the fluctuations presented earlier in that section are not present and the performance of the system scales linearly with the number of cores.

### B. The effects of HyperThreading

Most Intel CPUs support the HyperThreading [16] technology, which allows a CPU core to share its computing resources between two threads, thus appearing as two virtual cores to the operating system. On Linux the CPU cores are ordered such that the second halves of the CPU cores are the hyperthreads of the first half of the cores, in the same order. We tested this experiment over PC4, which supports HyperThreading technology.

We repeatedly ran two simultaneous instances of *stress-ng* with one stressor process each for 20 seconds, as part of the KVM/QEMU-based measurement sets (see Section IV.A). Fig. 10 shows the measured CPU frequencies and the number of operations completed for various setups: only one stressor, both on the same core, on different cores, on the two hyperthreads of the same core. If both physical cores are loaded, the CPU frequency decreases by 100 MHz, which shows in the per-thread performance, but even in this case the CPU runs well above its nominal frequency. Running two stressors on the two hyperthreads of the same core yields higher performance than running them on the same logical core, but it is nowhere near the performance we get when using two separate cores.

Thus, HyperThreading can indeed improve the performance of parallel computations beyond the number of physical CPU cores, but it is more useful in improving the responsiveness on a desktop PC than increasing the computing power of a server.

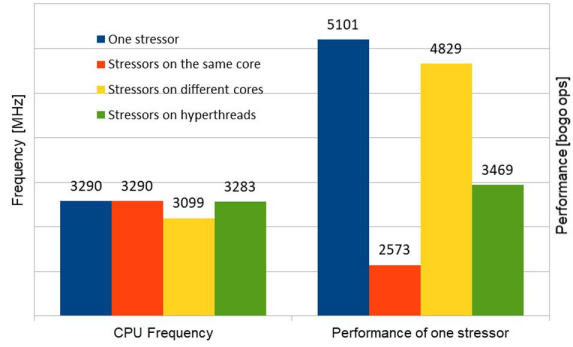



Fig. 10. HyperThreading results

Summarizing, if the monitoring process runs on the same CPU core as the monitored application, but on the other hyperthread, it can detect the load of the application while generating less interference than running on the same hyperthread. Of course, in a virtualized environment the processes running on the guest have no knowledge about HyperThreading of the host CPU; thus, exploiting it is usually not feasible.

### C. The effects of different CPU architectures

The brief tests shown in this section already illustrates the dependence of CPU performance on the CPU architecture and setup.

Our measurements were taken on multiple different computers, but we were not able to cover every possible architecture. For example, AMD CPUs are known to scale the frequencies of the cores more independently of each other than Intel CPUs, and when there is more than one CPU in the machine, those also scale their frequencies independently of each other. These properties may affect the sensitivity of the sidecar measurements negatively. Heterogeneous architectures exist too: in the ARM world the so called *big.LITTLE* architecture is very popular: depending on the workload a low power or a high-performance CPU core may execute the task. In the future it might be worth investigating the possibility of using sidecar measurements on such architectures.

## VI. CONCLUSION

In this paper we presented a measurement-based evaluation of the sidecar concept, aiming at evaluating the telecom application performance in a virtualized environment under dynamic load conditions. We considered several virtualization technologies and provided a quantitative analysis of the scenario.

According to our results the sidecar concept is viable. There is a correlation between the performance of the measurement application running in the sidecar and the resource usage of the main application running in a different VM or container. A good property of this measurement method is that the best sensitivity is achieved when the measurement application applies only slight load on the

system, thus creating low interference with the main application. The downside of this method is that it has low sensitivity when the main application is near full load, thus it cannot accurately predict an overload event. Running these measurements in a virtualized environment also adds challenges, as the visible resources not necessarily align with the resources that are physically available on that system.

## ACKNOWLEDGMENT

The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications)".

The authors thank the valuable help, motivation and technical guidance of Attila Gál and Olga Papp from Ericsson Hungary. We also thank the help of László Sári, who supported us in setting up the measurement environment.

## REFERENCES

[1] John, W., Moradi, F., Pechenot, B. and Sköldström, P., "Meeting the observability challenges for VNFs in 5G systems," *IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp. 1127-1130, 2017. DOI: 10.23919/INM.2017.7987445

[2] Burns, B., "How Kubernetes Changes Operations,"; *login:* The USENIX magazine, Vol. 40(5), 2015.

[3] Kernel Virtual Machine homepage – https://www.linux-kvm.org/page/Main_Page

[4] QEMU homepage – https://www.qemu.org/

[5] Libvirt, the virtualization API homepage – https://libvirt.org/

[6] Introduction to control groups (cgroups), RedHat documentation, https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/6/html/resource_management_guide/

[7] Namespaces - Overview of Linux namespaces, Linux Programmer's Manual, http://man7.org/linux/man-pages/man7/namespaces.7.html

[8] Docker homepage – https://www.docker.com/

[9] Kubernetes homepage - https://kubernetes.io/

[10] Configure Quality of Service for Pods, Kubernetes documentation, https://kubernetes.io/docs/tasks/configure-pod-container/quality-service-pod/

[11] Luong, D.H. et al., "Predictive Autoscaling Orchestration for Cloud-native Telecom Microservices," *2018 IEEE 5G World Forum (5GWF)*, pp. 153-158, 2018. DOI: 10.1109/5GWF.2018.8516950

[12] Van Rossem, S. et al., "Automated monitoring and detection of resource-limited NFV-based services," *2017 IEEE Conference on Network Softwarization (NetSoft)*, 2017. DOI: 10.1109/NETSOFT.2017.8004220

[13] Kraft S, Pacheco-Sanchez S, Casale G, Dawson S., "Estimating service resource consumption from response time measurements," *4th International ICST Conference on Performance Evaluation Methodologies and Tools*, pp. 1-10. 2009. DOI: 10.4108/ICST.VALUETOOLS2009.7526

[14] Khan M, Jin Y, Li M, Xiang Y, Jiang C., "Hadoop performance modeling for job estimation and resource provisioning," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 27(2), pp. 441-454, 2015. DOI: 10.1109/TPDS.2015.2405552

[15] stress-ng homepage – https://kernel.ubuntu.com/~cking/stress-ng/

[16] Marr et al., "Hyper-Threading Technology Architecture and Microarchitecture," *Intel Technology Journal*, 2002.

**Csaba Simon** obtained his PhD degree at Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics and he is working at the same Department since 2001. He is a member of the Balatonfüred Student Research Group. His research interests are mostly related to 5G systems and virtualization, IP QoS, peer-to-peer communications and network and service management. He was involved in several national and international research projects, covering his resarch topics. He is an active member of the Scientific Association for Infocommunications, Hungary, organising national conferences and being a contact for international relations and of the Sister and Related Societies Board at the IEEE ComSoc. He is the member of the International Working Group of the 5G Coalition, Hungary.

**Markosz Maliosz** received his PhD (2006) and MSc (1998) degrees in Computer Science from BME. He is a member of the Balatonfüred Student Research Group. He has participated in several national (OTKA-NKTH, TÁMOP, NFÜ) and EU-funded research projects (STREP, CELTIC, 5G PPP) and also worked in bilateral cooperation projects with Ericsson and Telia Research. His current research activity covers network virtualization and optimization focusing on industrial and cloud networking.

**Miklós Máté** received his MSc (2007) and PhD (2019) degrees in electrical engineering in the field of infocommunication systems at Budapest University of Technology and Economics (BME), Hungary. He is a research engineer in the High-Speed Networks Laboratory at the Department of Telecommunication and Media Informatics, BME. His research interests include intelligent transportation systems, distributed networks, and cloud technologies.

**Dávid Balla** is a PhD student at the University of Technology in Budapest, and also follows the PhD courses of the EIT Digital Doctoral School. He is a member of the Balatonfüred Student Research Group. He works at the High Speed Networks Laboratory at the university, and he is also the member of the research team at Ericsson Hungary. His main research topics are the physical and the software layers of cloud systems. During his master studies he worked with RDMA based interconnections and now he is dealing with Function as a Service and container based virtualization technologies.

**Kristóf Torma** graduated Budapest University of Technology and Economics in 2019. He is a member of the Balatonfüred Student Research Group. He joined the Faculty of Electrical Engineering and Informatics in 2020. His current research interest are cloud and container-based systems and their scaling behaviors, as well as scaling of IoT systems in Kubernetes.

# Amplified spontaneous emission based quantum random number generator

Ádám Marosits[1], Ágoston Schranz[2], and Eszter Udvary[3]

*Abstract*— **There is an increasing need for true random bits, for which true random number generators (TRNG) are absolutely necessary, because the output of pseudo random number generators is deterministically calculated from the previous states. We introduce our quantum number generator (QRNG) based on amplified spontaneous emission (ASE), a truly random quantum physical process. The experimental setup utilizes the randomness of the process. In this system, optical amplifiers (based on ASE) play the major role. The suitable sampling rate is selected in order to build the fastest generator, while avoiding the correlation between consecutive bits. Furthermore, the applied post-processing increases the quality of the random bits. As a results of this, our system generated random bits which successfully passed the NIST tests. Our real-time generation system – which is currently a trial version implemented with cheap equipment – will be available for public use, generating real time random bits using a web page.**

**Index Terms— quantum random number generator, amplified spontaneous emission, sampling rate, real-time generation**

## I. Introduction

Nowadays, there is an ever increasing demand for random numbers in communication and cryptography. The applications of random numbers include symmetric key cryptography, Monte Carlo simulations, protection of transactions, and key distribution systems, which will be more significant in the age of quantum computers. In order to generate true random bits (TRB), quantum random number generators (QRNGs) need to be implemented. Pseudorandom number generators (PRNGs) are widespread; they are cost-efficient because they algorithmically create seemingly random numbers, but they are deterministic, therefore these numbers cannot be declared as truly random. There are some random number generators, which sample complex physical processes, but with suitable measurements others can obtain the same numbers. Nevertheless, the randomness of quantum mechanics can provide high bit generation rates. Some quantum process based generators, for instance the radioactivity based QRNG, come with several serious problems: for example, the radiation is only enough just for a few detections per second, decreasing the generation rate. Moreover, we need huge quantities of radioactive materials, for which serious security arrangements need to be implemented. There are different possible processes for random number generation (e.g. the noise of chaotic circuits or the Brown-motion of particles), but it is not possible to generate high bit generation rates using these phenomena. We can differentiate between optical based QRNG systems, too. The first group is that of is the branching path generators, when the photon goes to a semi-transparent mirror that transmits it along one of the paths. At the end of both paths there is one detector, and the number of the detector signalling the arrival of a photon determines the value of the bit. The semi-transparent mirror is essentially a Hadamard gate, and at the end of the system, the value of the bit is 0 in 50%, and 1 in 50%. The second group are the photon counting generators. In this case, we count photon arrivals in a fixed-length time window, and we can decide on the value of the bit with a predetermined method. The third group is that of the time-of-arrival generators: random bits are generated based on the fluctuation of the time difference between photon arrivals. It is similar in principle to radioactivity based generators, but it is much more secure, since photons are used instead of particle radiation.

There is an another group of QRNGs that utilizes the randomness of amplified spontaneous emission (ASE) in order to generate random bits. The earliest proposal splits the signal into orthogonally polarized components with a polarization splitter, and calculates the difference between the independent polarization components; another one uses a balanced power splitter and tunable delay to symmetrize the intensity-fluctuation [1]. In order to not limit the generation rate, some earlier setup operates with optical filters, which have higher bandwidth than the receiver, to avoid its saturation. Some authors digitized the unfiltered, unamplified intensity-fluctuation at 16/32 bits from ASE sources. They got high bit generation rates and reduced the correlations by discarding several MSBs [2,3]. Several authors [2,3,4,5] used XOR post-processing methods to reduce short-term correlations. One article mentioned that the signal from a SLED (having a wide quasi-constant spectrum) is split and compared to the reference level to generate random bits [5].

In this paper, we present a QRNG that is based on amplified spontaneous emission. In the following sections, we discuss the theoretical background, the system, the suitable sampling rate selection, the success of post-processing and real-time bit generation. Our experimental setup operates with optical filters (CWDM, providing higher bandwidth than other standards) in order to avoid the saturation in the receiver. Furthermore, the above mentioned XOR post-processing method is applied to reduce short-term correlations. The signal is digitized at 1 bit, so that the quality of randomness will be easily investigated. If we compare the achieved 4 Gbps bit generation rate with the previous setups, we could find several faster implementations (the minimum rate was 2.5 Gbps [1], the maximum was 1.6 Tbps [3]), but digitizing at more bits and discarding several MSBs to reduce correlations, our setup could potentially achieve significantly higher bit generation rates.

## II. Theoretical Background

It is necessary to investigate the theoretical background of the phenomena in our setup, so that the generator can work as intended, and problematic operation can be avoided. Here we discuss these aspects in detail.

### *A. Amplified spontaneous emission*

Optical fiber amplifiers used in optical communications operate based on the effect of stimulated emission [6]. If an atom is in an excited state, it may, after some time, spontaneously decay into a lower energy level, releasing energy in the form of a photon. This process is called

[1,2,3] Department of Broadband Infocommunication and Electromagnetic Theory, Budapest University of Technology and Economics, Hungary
[1,2] BME Balatonfüred Student Research Group, Budapest University of Technology and Economics, Hungary
[1,2,3] E-mail: {marosits.a, schranz, udvary.eszter}@hvt.bme.hu

DOI: 10.36244/ICJ.2020.2.2

spontaneous emission [7]. However, it is also possible that the photon emission is stimulated by incoming photons, if these photons have suitable energy. This process is called stimulated emission. In that case the two photons in the output have identical properties. For stimulated emission to dominate over other types of light-matter interaction, population-inversion is required. It means that the population of particles is higher in the upper energy level than in the lower energy level. In many cases, it is achieved by optical pumping. If population-inversion exists, some of the particles from the excited state return spontaneously to the ground state. The photons, that are derived from spontaneous emission, may participate in stimulated emission; therefore, the optical amplifier amplifies its noise, too. The lack of input signal has several advantages: we don't have to filter the deterministic component and the accumulated energy is used to amplify the photons from spontaneous emission. This process is called amplified spontaneous emission (ASE) [8]. The emitted photons have random properties – for instance frequency –, so the amplified sum of the individual electric fields appears at the output as a swiftly fluctuating noise. The parameters of these photons don't correlate with the parameters of the signal photons. ASE cannot be described with classical electrodynamics; it is a quantum physical process. The generator – based on ASE – can generate true random bits using a method, where the measured intensity-fluctuation is compared to the mean, or in our case to the median (above the median a bit "1" is assigned to the sample, below the median a "0"). The bit generation rate is restricted by the device with the narrowest bandwidth, usually the detector [9].

### *B. ASE sources*

Several types of devices can be used as ASE sources. In case of semiconductor optical amplifiers (SOA) [10], the population-inversion is achieved by current injection. Without any input signal, the SOA uses the accumulated energy to amplify its own noise originating from spontaneous emission. In this mode, the SOA functions as an ASE source.

Erbium-doped fiber amplifiers (EDFA) [11] are optical fiber amplifiers, where the necessary energy is provided by a laser diode. The pumping laser provides the population inversion. These lasers generally operate at 980 nm or 1450 nm. After the excitation, there is a quick non-radiative transfer of the ions to a metastable energy level, from where they may return the lower energy level, releasing a photon with a wavelength around 1550 nm by stimulated emission. In this case we can speak about a quasi-three-level transition [12]. This is an intermediate situation, where the lower energy level is so close to the ground state, because there is an appreciable population in thermal equilibrium at the operating temperature. The particles from the metastable state may return to this state by emitting lower energy photons. This energy loss is called reabsorption loss.

The EDFA in our laboratory was a part of a DWDM infocommunication system, where energy saving is an important aspect; therefore, it doesn't turn on at low input powers (< -29 dBm). Consequently, this equipment is not used as an ASE source, but it provides a high gain, so it is suitable for amplification in our system.

### *C. Saturation of the optical-electrical converter*

The lightwave converter is responsible for converting optical intensity to electrical voltage. The device is essentially a photodiode with a transimpedance amplifier (TIA). In our system, the lightwave converter provides the connection between the optical system and the oscilloscope. The photodiode is a photosensitive diode operating based on the photoelectric effect. The incoming photons are absorbed, generate a photocurrent, and this photocurrent is converted to voltage by the TIA. Consequently, the electric voltage is proportional to the optical intensity; more precisely, to the square of the optical field strength. The saturation of the lightwave converter can cause false measurement results and the equipment may be damaged if the incoming power is too high. Saturation happens when the increasing optical power cannot increase the voltage with the same linearity as before. The power–voltage characteristic of our lightwave converter can be seen in Figure 1. The P–V characteristic is linear below 5 dBm, but above this values it does not increase at the same rate. This is the saturation power; therefore, we have to maximize our system's optical output power under 5 dBm.

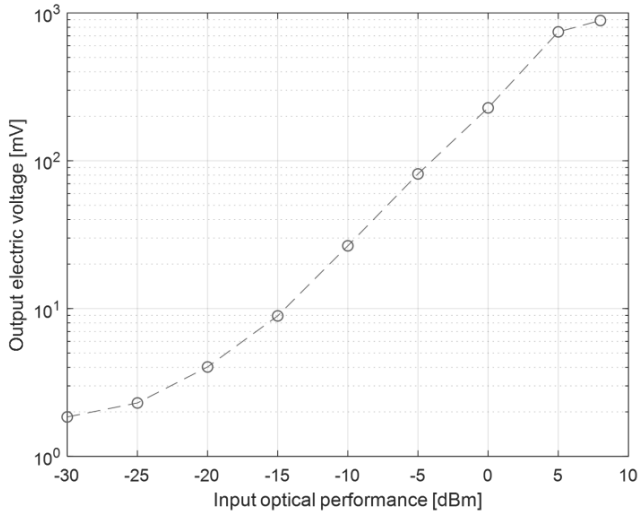



Figure 1. The P-V characteristic of the lightwave converter.

### *D. Asymmetric intensity-fluctuation*

The asymmetry of the measured intensity-fluctuation has caused a significant amount of problems during measurements, so the reason behind it and the solution against it need to be clarified. The signal appearing at the output of the lightwave converter can be described as a random variable following a gamma distribution.

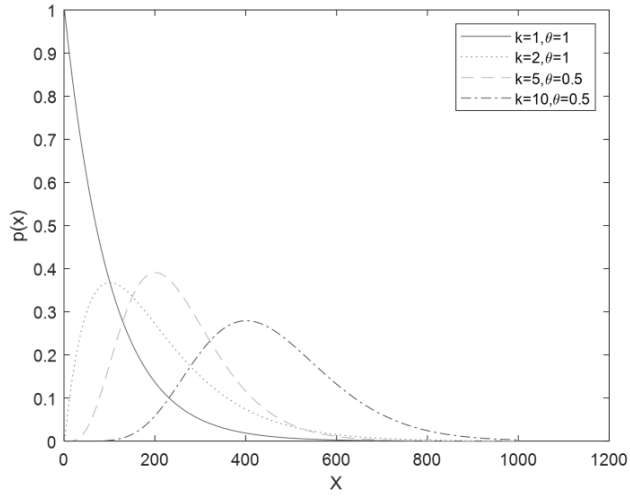



Figure 2. The gamma distribution's probability function with different parameter values for k and Θ.

The gamma distribution has an asymmetric probability density function (PDF). The PDF (Figure 2.) – using the shape-scale parametrization – can be written as

$$f(x,k,\theta)=\frac{x^{k-1}\cdot e^{-\frac{x}{\theta}}}{\theta^{k}\cdot\Gamma(k)},$$

where $\Gamma(k)$ is the gamma function, k is the shape parameter, θ is the scale parameter. For large k, the asymmetric gamma distribution converges to a symmetric normal distribution with mean $\mu = k\cdot\theta$ and variance $\sigma^2 = k\cdot\theta^2$. The skewness of the gamma distribution only depends on the shape parameter, and is equal to $2/\sqrt{k}$ and inversely proportional to the square root of the optical intensity. Consequently, large optical intensities provide quasi-symmetric distributions, which is beneficial, if we would like to achieve a uniform distribution of 0 and 1 bits. The large optical intensity results in a large mean of the intensity-fluctuation. In Figure 3., there are two color scale displays of intensity-fluctuations with different means. The difference between an asymmetric and a quasi-symmetric distribution is clearly visible.

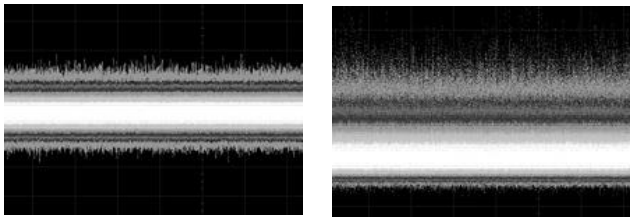


Figure 3. Left: the color scale display of the intensity-fluctuation with high DC voltage (631 mV), right: the color scale display of intensity-fluctuation with low DC voltage (33 mV). The first follows a highly symmetric distribution, while the second is highly asymmetric.

## III. EXPERIMENTAL SETUP

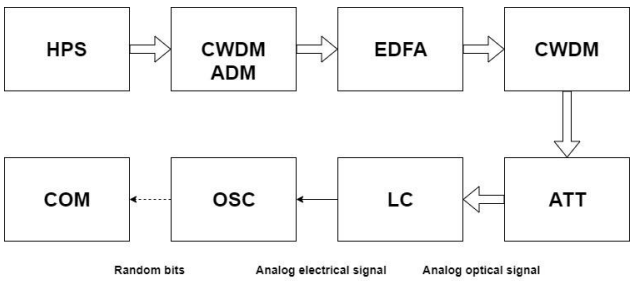



Figure 4. The experimental setup's block diagram.

The experimental setup uses a Perkin-Elmer High Power Source (HPS) as the source of ASE. It has a similar spectrum to the SOA, but it has higher noise power near 1550 nm (the difference is 10 dB), where our system operates. The two spectra can be compared in Figure 5.

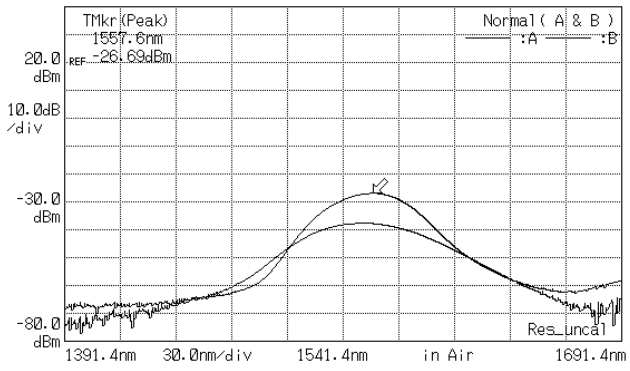



Figure 5. Comparison between the SOA and the HPS spectrum (the latter has a higher peak).

The fluctuation amplitudes are not high enough to generate true random numbers, therefore they are amplified by an EDFA. We apply a CWDM add-drop multiplexer as prefilter. The prefilter cuts off the unwanted sideband components, so that the EDFA does not amplify the whole band. It means that the accumulated energy is used to amplify in a narrower range, causing higher suppression of the EDFA's own noise. Here we use the CWDM standard, because it has a higher bandwidth than DWDM or other filters, so that the input power is large enough to turn on the EDFA. The filter has a bandwidth of 19.2 nm, and the insertion loss is practically negligible. The filtered and unfiltered spectra are compared in Figure 6.

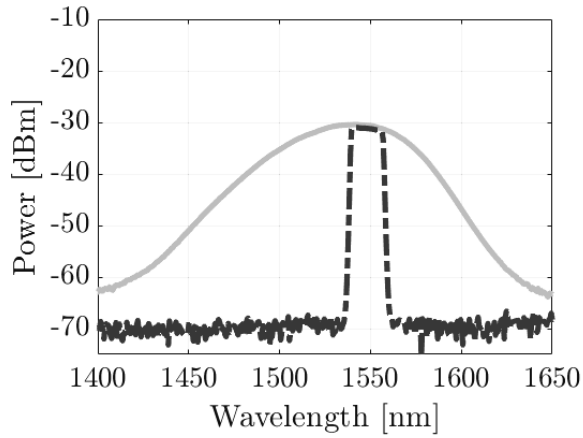



Figure 6. The unfiltered (cont.) and filtered (dashed) HPS spectrum.

The EDFA amplifies the signal significantly. The maximum of the optical power is at 1542.6 nm (-13.49 dBm). The EDFA's own noise is suppressed by 22.4 dB. It is 7.4 dB higher compared to the case when the SOA is applied as the ASE source. However, the huge total power causes saturation in the optoelectrical converter. The EDFA amplifies everything within its gain spectrum, so using another filter at the end of the system is inevitable. We use a CWDM filter again, with the purpose of providing enough optical intensity that the gamma distribution converges to the symmetric normal distribution. This filter has 2.5 dB insertion loss around 1550 nm. The output power after the second filter is 8.1 dBm, therefore we use an attenuator with around 3.5 dB attenuation. The detected power at the output of the optical system (4.6 dBm) is high enough to avoid an asymmetric distribution, but low enough to avoid saturation. The amplified and filtered spectra are shown in Figure 7.

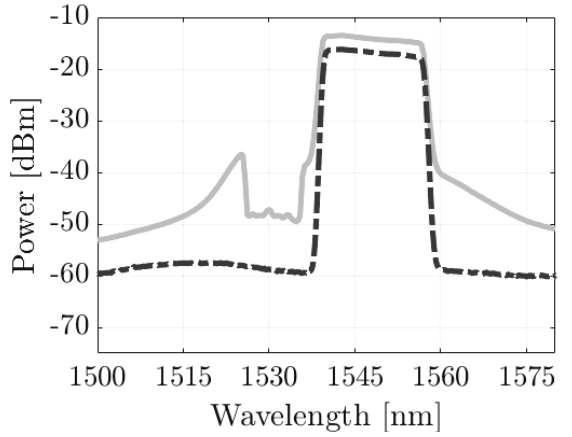



Figure 7. The HPS spectrum amplified by the EDFA (continuous) and the filtered optical spectrum after amplification (dashed).

An optical-electrical converter (called a lightwave converter) is used to convert optical intensity to electrical voltage. The color scale display of the detected intensity-fluctuation is shown in Figure 8. It has an average of 631.87 mV and peak-to-peak voltage of 250.07 mV. It is clearly visible that it is nearly symmetric, being beneficial in terms of randomness quality.

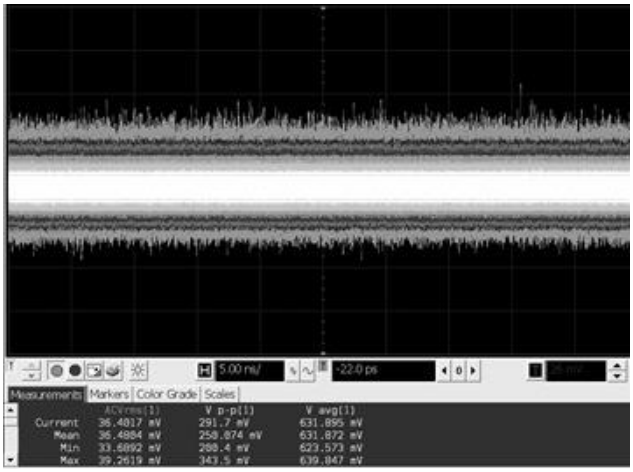



Figure 8. The detected intensity-fluctuation.

## IV. SAMPLING AND POST-PROCESSING

The electrical signal is digitized and stored by an oscilloscope and processed offline by Matlab [13]. The median compared samples provide theoretically uniformly distributed 0 and 1 bits. Due to the fact that we compared to the median, some values coincide with it. To not lose any bits, we added 50 μV to these samples. It causes a deviation from the uniform distribution, but this intentional error highlights the differences between sampling rates. We assigned just one bit to the values, because the quality of random numbers in different rates shows higher contrast. The rates are chosen so that they cover a wide range around the analog bandwidth (the scope's 8 GHz bandwidth is lower than the photoreceiver's, limiting the bit generation rate). The values were 0.1, 0.2, 0.5, 1, 2, 4, 10 and 20 GSa/s. Unfortunately, the sampling with 8 GSa/s was not supported by the oscilloscope. $10^9$ samples are collected for each sampling rate and 1000 bitstreams were created from these that consist of $10^6$ bits. All bit streams are then subjected to randomness testing. A decrease in the quality of randomness is expected with increasing sampling rate, because it causes short term correlations, especially for those higher than 8 GSa/s. For the evaluation of randomness, we used the NIST (National Institute of Standards and Technology) [14] test suite that contains 15 different tests. All tests generate a p-value, representing the probability accepting the null hypothesis that the bitstream is random. This p-value needs to be higher than the significance level, and p-values should be uniformly distributed. Nevertheless, QRNGs should generate all sequences of a given length with the same probability. It means that the generator is expected to fail sometimes (at least once), but no more than 20 times out of 1000 sequences. This is the condition of the success. The results of testing are shown in Figure 9. There is a tendency as the sampling rate grows, decreasing the number of the successful tests (or keeping the value). There is just one exception: in case of 0.2 GSa/s there are more successful tests than in case of 0.1 GSa/s.

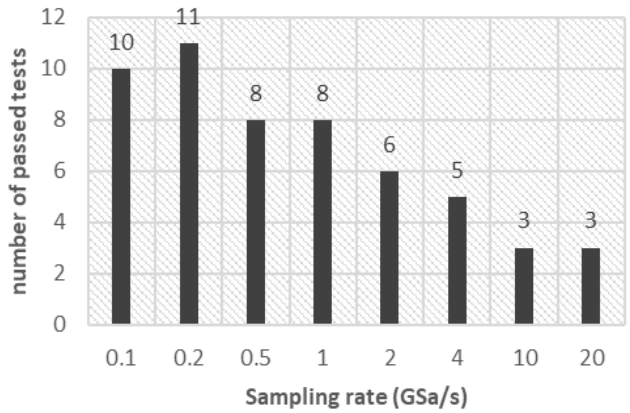



Figure 9. The number of passed NIST tests at different sampling rates.

In order to reduce correlations in the raw bit streams and increase the quality of randomness, we use a post-processing method. It was a simple XOR technique: we take the XOR operation of the original sequence with a delayed sequence (we apply a 20 bit delay). It drastically increased the number of passed tests, as it is clearly visible in Figure 10. This method reduced the short-term correlations. After post-processing, the bit sequences at 0.1 GSa/s, 0.5 GSa/s and 1 GSa/s passed everything. Two of them (0.2 GSa/s and 4 GSa/s) passed all except one test and one of them (4 GSa/s) passed all except two tests. The two sequences with sampling rates higher than the oscilloscope bandwidth (8 GHz) also passed more tests than before the post-processing, but despite of it, they are unsuitable for the single-bit generation in our system.

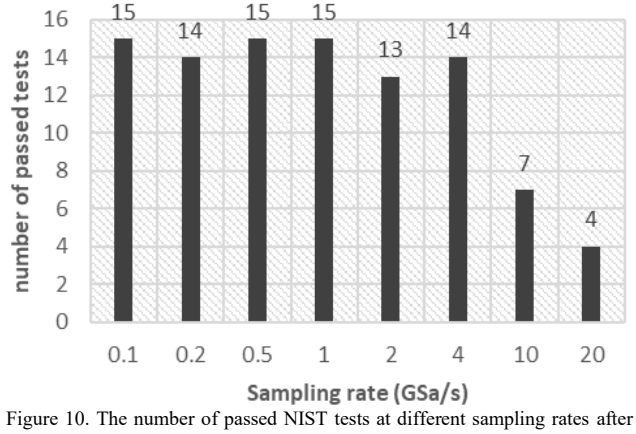



Figure 10. The number of passed NIST tests at different sampling rates after post-processing.

We can conclude that the randomness is heavily depending on the sampling rate. According to expectations, with the sampling rate being below the analog bandwidth, the quality of the randomness is better than above it. However, it means a decrease in the bit generation rate. It is foreseeable that the self-delayed XOR method can increase drastically the quality of randomness below the analog bandwidth. The optimal choice is 4 GSa/s for us, which is the highest sampling rate with just one unsuccessful test (suitable post-processing can eliminate this failure). There are some additional opportunities. Firstly, the mean will be a better comparison level than the median, because we can avoid the intentional comparison failure. Furthermore, we can increase the XOR delay, which could reduce the short-term correlations even more. Besides, we can assign more bits to one sample. Although it causes additional correlations, these can be reduced by discarding several MSBs.

## V. REAL-TIME GENERATION

As of now, our website is a trial version of the future online interface, which will operate with an electrical level shifting circuit, and it will be available for public use, but the current version is able to generate bits. This version only uses cheap equipment that limits the bit generation rate. The website runs on a Raspberry Pi. The Pi has no built-in ADC; the converteor runs on a distinct ESP8266 microcontroller. The hardware arrangement is visible in Figure 11.

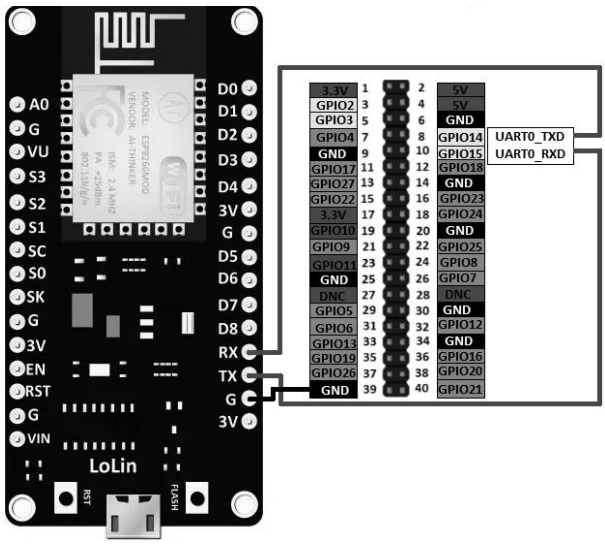



Figure 11. Block diagram of the real-time RNG.

The signal from the optical-electrical converter is sampled by a 10-bit ADC operating between 0 V and 3.3 V. The communication between the microcontroller and the Pi is on UART.

The ESP8266 is programmed in C language. If the serial port becomes active, the equipment reads the number of the requested bits. Then the ESP8266 fills an n-bit array from the analog input. The equipment repeats this method every time the Pi requests bits, so the reuse of samples is excluded. These values are between 0 and 1024. We compare the values to the mean and refill the n-bit array with 0 and 1 bits (bigger n means more precise average). Then the equipment creates a string with suitable size (it is equal to the number of the requested bits) and this string is transmitted to the Pi on a serial port.

The Raspberry Pi is programmed in HTML, CSS, PHP, Python and SQL languages. The website is created by a PHP file, where the number of bits to receive can be specified. The "Generate" button creates an SQL record, which requests bits from the ESP8266. The requesting, processing and ready state have different flags in the SQL table. Then the *.php* file starts the *.py* file. This file selects the record (which is in a sending state) from the SQL table and requests the number of the requested bits. This number is sent to the ESP8266 by the *.py* file, and while the microcontroller sends back the appropriate amount of bits, the file creates a SQL record in processing state. Finally, the *.php* file writes the bits onto the webpage.

| id | src | dst | msg | number | bitnumber | sending | processing | ready |
|---|---|---|---|---|---|---|---|---|
| 326 | WEB | SRV | GetRndBits | 0 | 7 | 0 | 0 | 1 |
| 327 | SRV | WEB | SetRndBits | 1011101 | 7 | 0 | 0 | 1 |
| 328 | WEB | SRV | GetRndBits | 0 | 8 | 0 | 0 | 1 |
| 329 | SRV | WEB | SetRndBits | 10111 | 8 | 0 | 0 | 1 |
| 330 | WEB | SRV | GetRndBits | 0 | 7 | 0 | 0 | 1 |
| 331 | SRV | WEB | SetRndBits | 1011111 | 7 | 0 | 0 | 1 |

Figure 12. The registration system.

There are some additional opportunities to develope. Currently a level shifting circuit is in the development phase. This circuit removes the mean from the measured intensity-fluctuation, then extends the fluctuation levels to the whole range of the ADC with a multiplier circuit and offset voltage. The advantage of this circuit is that there would be no need to calculate the average, because the system provides bits, where the comparison threshold is the half-range of the ADC. Another opportunity is to operate the real-time bit generating system as a real webserver. Currently the system just operates on a local network, but the access is expansible for everyone. Finally, higher bit-generation rates are achievable with sampling for instance on 16-32 bits. By discarding several MSBs, the problems of oversampling are reducible.

## VI. CONCLUSION

The QRNG presented in this paper is based on ASE, a truly random quantum mechanical process. The ASE noise is filtered with the drop channel of a CWDM add-drop multiplexer. An EDFA is used to amplify this signal to avoid the asymmetry of the intensity distribution. After the EDFA, a second CWDM filter (and an attenuator) is applied to reduce the power in order to avoid the saturation of the receiver. After the optical-electrical converter, the intensity-fluctuation was single-bit sampled offline with Matlab. The effects of sampling rate were tested, and the conclusion was that below 8 GSa/s (the analog bandwidth of oscilloscope) the sampling rate is acceptable to generate random bit sequences. However, the application of post-processing is inevitable to increase the quality of randomness. The optimal sampling rate is 4 GSa/s with a suitable post-processing method, because it is the highest available bit generation rate below 8 GSa/s. Furthermore, the developed webserver running on a Raspberry Pi provides the opportunity to generate bits real-time with a registry system. Further improvements might be done to the system, so that the bit generation can be effectively reduced to a zero-level comparison, and the real-time bit generation rate can be increased to be closer to the what could be theoretically possible.

## ACKNOWLEDGMENT

The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications).

## REFERENCES

[1] L. Li, A. Wang, P. Li, H. Xu, L. Wang, and Y. Wang, „Random bit generator using delayed self-difference of filtered amplified spontaneous emission," IEEE Photonics Journal, vol. 6, no. 1, pp. 1–9, 2014.

[2] A. Argyris, E. Pikasis, S. Deligiannidis, and D. Syvridis, „Sub-tb/s physical random bit generators based on direct detection of amplified spontaneous emission signals," Journal of Lightwave Technology, vol. 30, no. 9, pp. 1329–1334, 2012.

[3] Y. Liu, M. Zhu, B. Luo, J. Zhang, and H. Guo, „Implementation of 1.6 Tbs-1 truly random number generation based on a super-Á. Marosits et al.: Amplified spontaneous emission based quantum random number generator luminescent emitting diode,"Laser Physics Letters,vol.10, no. 4, p. 045001, 2013.

[4] C. R. Williams et al., "Fast physical random number generator using amplified spontaneous emission," Optics express, vol. 18, no. 23, pp. 23584–23597, 2010.

[5] X. Li et al., "Scalable parallel physical random number generator based on a superluminescent led," Optics letters, vol. 36, no. 6, pp. 1020–1022, 2011.

[6] R. Paschotta, article on 'stimulated emission' in the Encyclopedia of Laser Physics and Technology, 1. edition October 2008, Wiley-VCH, ISBN 978-3-527-40828-3

[7] R. Paschotta, article on 'spontaneous emission' in the Encyclopedia of Laser Physics and Technology, 1. edition October 2008, Wiley-VCH, ISBN 978-3-527-40828-3

[8] R. Paschotta, article on 'amplified spontaneous emission' in the Encyclopedia of Laser Physics and Technology, 1. edition October 2008, Wiley-VCH, ISBN 978-3-527-40828-3

[9] M. Herrero-Collantes and J. C. García-Escartín, "Quantum random number generators," Reviews of Modern Physics, vol. 89, no. 1, p. 015004, 2017.

[10] R. Paschotta, article on 'semiconductor optical amplifiers' in the Encyclopedia of Laser Physics and Technology, 1. edition October 2008, Wiley-VCH, ISBN 978-3-527-40828-3

[11] R. Paschotta, article on 'erbium-doped fiber amplifiers' in the Encyclopedia of Laser Physics and Technology, 1. edition October 2008, Wiley-VCH, ISBN 978-3-527-40828-3

[12] R. Paschotta, article on 'four-level and three-level gain media' in the Encyclopedia of Laser Physics and Technology, 1. edition October 2008, Wiley-VCH, ISBN 978-3-527-40828-3

[13] Á. Schranz, Á. Marosits, and E. Udvary, „Effects of sampling rate on amplified spontaneous emission based single-bit quantum random number generation," in 2019 21st International Conference on Transparent Optical Networks (ICTON), pp. 1–4, IEEE, 2019.

[14] A. L. Rukhin et al., „A statistical test suite for random and pseudorandom number generators for cryptographic applications," tech. rep., National Institute of Standards & Technology, Gaithersburg, MD, United States, 2010. Spec. Pub. 800-22, Rev. 1a

**Ádám Marosits** is currently a BSc student in electrical engineering at the Budapest University of Technology and Economics (BME, Budapest, Hungary) at the Department of Broadband Infocommunications and Electromagnetic Theory. He is a member of the Balatonfüred Student Research Group. His research interests include quantum communication and quantum random number generation, furthermore the optimization with quantum annealing.

**Ágoston Schranz** received his BS and MS degrees in electrical engineering from the Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2015 and 2017, respectively. Currently, he is working toward his PhD in electrical engineering at the Department of Broadband Infocommunications and Electromagnetic Theory. He is a member of the Optical and Microwave Telecommunication Laboratory and the Balatonfüredi Student Research Group. His research interests include optical communications, quantum key distribution, and quantum random number generation.

**Eszter Udvary** (M'98) received the PhD degree in electrical engineering from Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2009. She is currently an Associate Professor at BME, Department of Broadband Infocommunications and Electromagnetic Theory, where she leads the Optical and Microwave Telecommunication Lab. Dr Udvary's research interests are in the broad areas of optical communications, include optical and microwave communication systems, Radio over fibre systems, optical and microwave interactions and applications of individual electro-optical devices. Her current research interests are future quantum and visible light communications.

# Optimisation of Optical Network for Continuous-Variable Quantum Key Distribution by Means of Simulation

David Kobor[1] and Eszter Udvary[2]

*Abstract*—**The unprecedented breakthrough in the field of quantum computing in the last several years is threatening with the exploitation of our current communication systems. To address this issue, researchers are getting more involved in finding methods to protect these systems. Amongst other tools, quantum key distribution could be a potentially applicable way to achieve the desired level of protection. In this paper we are evaluating the physical layer of an optical system realising continuous variable quantum key distribution (CVQKD) with simulations to determine its weak points and suggest methods to improve them. We found that polarisation dependent devices are crucial for proper operation, therefore we determined their most defining parameters from the point of operation and suggested extra optical devices to largely improve transmission quality. We also paid attention to polarisation controlling in these sort of systems. Our findings could be valuable as practical considerations to construct reliable CVQKD optical transmission links.**

**Index Terms:** quantum communication, quantum key distribution, CVQKD, optical network, simulation

## I. INTRODUCTION

Since the first great achievements of the eighties and nineties [1], [2], quantum information technology has been drawing increasing attention, and promising groundbreaking technical solutions. However, the rapid development of quantum computing does not only have unquestionable merits, but also poses significant security threats to our existing communication networks. Thus, it is unavoidable to come up with brand new methods to ensure undisrupted operation for the future. For example, some of the most widespread encryption algorithms are relying on the very assumption, that it's rather hard - on human scale impossible - to factor large prime numbers. As quantum computers are beginning to be commissioned, this will no longer to be impossible.

A very promising field of research to protect our communication networks has been quantum key distribution (QKD), which is looking to protect the most easily exploitable part of symmetric key encryption: the distribution of common key between the communicating parties. As soon as the exchange of the secret key is considered to be secure, the proceeding communication is safe. Quantum key distribution can be divided to three major categories: discrete-variable QKD, continuous-variable QKD and distributed-phase-reference QKD. In this paper the focus is on continuous-variable QKD (CVQKD). CVQKD offers the major advantage of not requiring any special, high-cost components, but might be built up using only conventional telecommunication devices [3]. This fact makes it relatively easy and straightforward to implement and measure test devices. To ensure the highest possible key rate all noise contributions of the link must be kept as low, as possible, regardless of the external or internal source. This the reason why CVQKD connections are in many cases realised over optical fibre, but there have also been efforts to establish a connection over free-space [4] and evaluating free space transmission [5].

In 2008 the European Integrated Project (SECOQC) team proposed a working CVQKD connection over 8 km of optical fibre, at 8 kbps key rate [6], [7]. The aim of Symmetric Encryption with QUantum key REnewal (SEQURE) project has been the same. They maintained quantum secured communication over 12 km of fiber at a maximum of 1 kbps key rate. The Budapest University of Technology and Economics (BME) have also started developing a setup for quantum key distribution to demonstrate its feasibility [8]. In the last couple of years great effort has been devoted to the difficulties of practical implementation. Researchers are looking for methods of extending the link range (e.g. with new protocols [9]), maintaining connection over different mediums [10], [11], and trying to optimise electrical or optical components of the complex system [12], [13], as well as giving better theoretical description of the employed devices [14]. In this paper we are taking a different approach and use classical optical system simulation (VPI Transmission Maker) to evaluate the optical layer of a CVQKD network in order to optimise the parameter choice, and come up with suggestions regarding the specific optical devices. Our goal is to conduct simulations prior to building the quantum link, to get an idea, what level of transmission quality might be expected from the system. CVQKD has been investigated from many points of view, the theoretical basis of this system has already been worked out in [8], but there has been very little discussion about the actual physical construction of such systems (for example how to choose the optical components and what to look after when assembling them). We are suggesting minor changes in the already proposed

[1]BME Balatonfüred Student Research Group, Budapest University of Technology and Economics, Hungary
[2] Department of Broadband Infocommunications and Electromagnetic Theory, Budapest University of Technology and Economics, Budapest, Hungary
[1,2] E-mail: david.kobor@edu.bme.hu; udvary@hvt.bme.hu

DOI: 10.36244/ICJ.2020.2.3

architecture, focusing on weak points in the design of the physical layer to improve on it to the highest extent possible. Our design considerations might not only be utilised for system comprising of discrete components, but could also be useful for integrated photonic chip design for CVQKD [15].

In Sec. 2 we are describing the operation of the system on the level of the optical network, list several important consideration that we kept in mind during the simulations. In Sec. 3 we detail the most important undesired mechanisms we identified that are affecting system performance. In Sec. 4 we give an overview of our polarisation controlling method and its impairments. Sec. 5 is to summarise our findings and draw the conclusions.

## II. System Simulation

In this section we describe the network we had implemented in the simulation environment, as well as the most important consideration we kept in mind before and during the simulations.

### A. Model

The block diagram of the system is depicted in Fig. 1. Extremely low energy impulses are used to communicate between the two parties (Alice and Bob), whose behaviour might be described using the laws of quantum physics. Basically, it is operating as a self-seeded homodyne optical transmission system with balanced detector structure, however certain modifications are applied to ensure flawless quantum operation. The transmitter is producing complex numbers coded in the quadrature of extra-low energy light impulses. Quantum operation is based on the sufficiently low energy of these pulses.

The optical carrier is produced on Alice's side by a laser, which is protected with an isolator, while the CW light is modulated with high extinction intensity modulator to get light impulses of certain periodicity. In ideal operation the laser light is assumed to be perfectly linearly polarised. In the figure this is marked with the red signal. The impulses are split in 90:10 ratio in two separate paths: the reference for homodyne detection and the signal carrying the actual information (modulated signal). 90% of power is going directly to the output of the transmitter (PBS 2), while the 10% is intensity and amplitude modulated and attenuated even further to reach the desired power level. The cumulative attenuation of the modulated path should be set in such way, that the transmitted packets contain at most a few hundred photons. This low transmitted power per packet ensures the overall security of the system. The path of the modulated signal is extended with an additional structure seen in Fig. 1. This signal is routed to the X polarised port of a polarisation beam-splitter (PBS 1), where X is the assumed polarisation state of the laser. This routes the signal to the common port of the PBS (XY of PBS 1), where it is passed through an optical delay line and is reflected by a Faraday-mirror (FM). The FM rotates the polarisation of the signal by $90^\circ$ and reflects it back to PBS 1 through the delay line again, where it is now exiting the other (Y polarised) port. Here it gets recombined with the reference signal, but now the impulses are not only separated in time, but are also orthogonal in polarisation (due to the effect of FM). These optical functions (delay and polarisation rotation) are applied to ensure minimal interference between the modulated and the reference signal while propagating on the long optical fibre connecting Alice and Bob.

During the propagation on the fibre the state of polarisation of the signal gets further rotated by an unpredictable amount. To overcome this we use a polarisation controller to reset the initial state we had on the transmitter output. This is a crucial point, because in the next we seek to separate the reference and the signal again. We use a polarisation beam splitter (PBS 3) for this function and successful separation requires perfectly set polarisation. The modulated signal gets directly to the balanced receiver. On the reference path the reference signal is routed to the very same structure what we had in the transmitter (PBS 4). This time the reference signal is rotated in polarisation and delayed, to be in the same time and polarisation state as the modulated signal, when it reaches the balanced receiver.

### B. Simulation Considerations

- We are using VPI Transmission Maker to build the model of the system depicted above. This software has been developed to evaluate conventional optical transmission links based on classical physical principles, therefore it is unable to take any quantum behaviour into consideration. However, we are not looking to demonstrate any complex key-sharing operation or any protocol level use. Our goal is to improve the optical layer of the key sharing system, by achieving the lowest noise and self-interference, which might be successfully done with classical methods.
- The architecture described above is the very same, that has been built at the Budapest University of Technology and Economics. We were using the parameters of the devices they employed based on their data sheets.
- The laser source is not perfectly linearly polarised. We didn't find any information about the exact grade of imperfection in the data sheet, but therefore we assumed a power ratio of 100 (20 dB polarisation extinction ratio) between the orthogonal polarisation axes.
- Polarisation management is utterly important throughout the whole system, however most of the data sheets don't detail polarisation dependent operation. Conventional beam splitters and isolators are not polarisation maintaining (PM) elements. There are available PM devices, but these also affect the polarisation, however their effect is much smaller than that of the conventional components. This contribution might be expressed with an angle of rotation. In case of all PM and non-PM devices we approximated a degree of rotation on the polarisation states to describe their behaviour. We are also assuming a worst-case situation, meaning that all rotations are performed in the same direction, the effect of the succeeding devices don't cancel out.

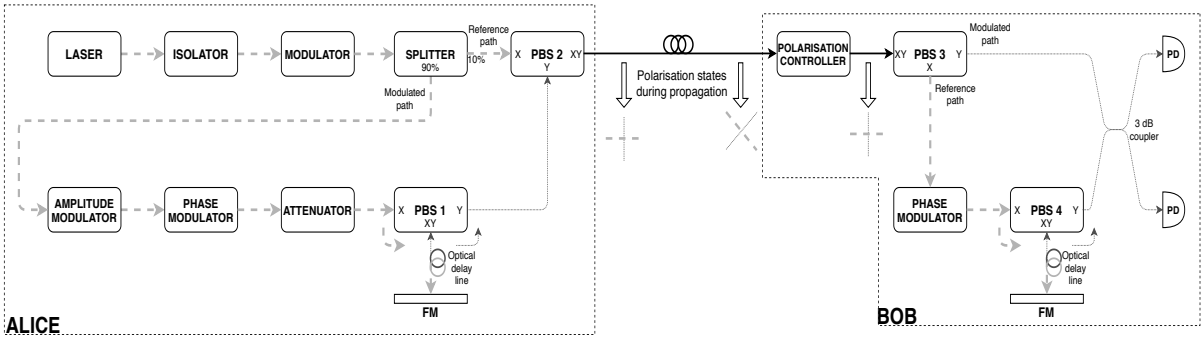



Fig. 1. Block diagram of CVQKD optical system; two orthogonal polarisation states marked with light grey dashed line (X) and dark grey dotted line (Y); the transmitter (Alice) produces the pulsed optical carrier, which is split in two paths of which one is modulated and has its polarisation rotated. The two paths are then combined and sent to the receiver (Bob), where the signal is split again based on polarisation and finally detected with a coherent receiver. PBS 4 is used to match the polarisation of signals in the two branches.

- Polarisation beam splitters (PBS) and combiners are employed for two purposes in our system. First we use them to separate and join the reference signal and the one carrying the actual information. This happens at the output of the transmitter and the input of the receiver. We also use PBSs for a different purpose: to construct circulator-like sub-systems to delay and change the state of polarisation of the signals. We are going to use the graphical representation of the PBS seen on Fig. 1 (PBS 1) for further explanation. In case of our circulator-like structures the light enters at one of the polarised ports (X) and exists the common port (XY). When it is reflected by the Faraday-mirror, it goes through XY again, but this time exits the device at the other polarised port (Y), because its polarisation has been modified. The manufacturer defines the polarisation extinction ratio (PER): when perfectly linearly X polarised light enters the common port (XY), its full power should be exiting the dedicated port (X). Due to the imperfections some X polarised light gets to the Y port, and PER tells us how significant this contribution is. This is crucial parameter in our application, and it must be modelled very carefully. There is another important parameter, which we could not find in any data sheet: the crosstalk between the singular (X and Y) ports. As we feed an optical signal to the X port, its full power is bound to exit on the common port (XY). But based on our experiences, there is a tiny fraction of light, that gets immediately reflected from the XY port and goes directly to Y by skipping the optical delay line and Faraday-mirror. According to our measurements this crosstalk should be in the domain of -60 dB, which is rather low, but still crucial in our CVQKD optical system. This had to be modelled very carefully.
- We didn't have any information regarding the polarisation dependent behaviour of phase and amplitude modulators, therefore we assumed ideal operation.

### C. Evaluation

We are treating the CVQKD optical system as a conventional transmission system, therefore we utilise classical methods of evaluation. For this purpose we utilised simple QAM and PSK modulation schemes and we took their error vector magnitude (EVM) as a descriptive metrics. Moreover, we have also taken the time domain waveforms produced by the simulator, so we can learn more about the nature of flaws and weaknesses during the operation.

## III. Simulation Results

1000 randomly generated symbols have been sent during the simulations, while we were looking at the EVM and waveforms. The definition of EVM might be seen below:

$$\mathrm{EVM}(\%) = \sqrt{\frac{P_{\mathrm{error}}}{P_{\mathrm{reference}}}} \times 100\% \qquad (1)$$

The symbols of reference for are known, the error vectors can easily be calculated from the simulation results. We have identified two important mechanisms that have a significant impact on transmission quality and need to be addressed. We are using this section to describe these mechanisms.

### A. Insufficient separation of reference and modulated signals

The complete system seen on Fig. 1 is constructed to provide the best possible separation between the reference signal (LO) and the modulated signal and minimise crosstalk. This is the reason for using Faraday-mirror and optical delay line. However, mainly due to the imperfections of the employed optical devices this is not always enough. Considering that the reference signal has a much higher power than the modulated signal, it is able to completely ruin the operation. Even if we use devices sold as polarisation maintaining, we can be sure that they are not performing perfectly, they have a certain degree of polarisation changing effect. The same goes for the laser, whose cross-polarisation suppression can never be infinite. The polarisation controller is only able to compensate the impairments caused by the fibre connecting Alice and Bob. This is why we have to be very considerate in the design phase when choosing the device parameters. Our splitter (seen in Fig. 1) is splitting the optical carrier in 90:10 ratio, where the 90% is the reference signal. At this point we assume, that the light is linearly polarised, but this is not true. In fact, after the splitting the optical carrier is going to be mainly polarised

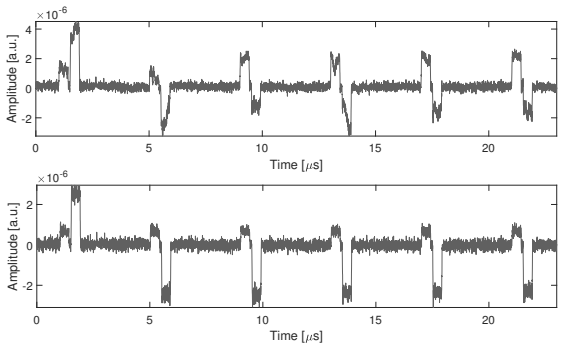



Fig. 2. Comparison of output waveforms without and with polariser on Alice's side

along the X axis on both routes, but a certain fraction is going to be orthogonal to it (Y). On the reference route this light is going to the output polarisation combiner (PBS2), while its X aligned component is desired, its Y aligned component is undesired. On the modulation route the 10% of power is further attenuated and transformed to be Y polarised (using the Faraday-mirror). Subsequently the modulated signal is routed to PBS 2 and recombined with the reference signal. The issue in that the modulated signal is Y polarised on purpose, while the X polarised reference signal also has a fraction of Y polarised power. These Y polarised contributions have about the same magnitude of power, because the modulated signal has been strongly attenuated, while the reference signal hasn't been. They propagate over the same medium connecting Alice and Bob, thus we must expect considerable interference. We have found and simulated several methods to avoid this.

- The first idea could be to insert a polariser after the laser source to improve its linearly polarised behaviour. According to our simulation this is not a desired method, because all subsequent devices also modify the polarisation, therefore when the light reaches the critical place (PBS 2), it is not going to be well polarised anymore.
- We have found the place in the system to use a polariser with the highest efficiency in minimising the interference of the modulated and reference signals. It is to be used between the Splitter and PBS 2 in the reference route, in order to decrease the power propagating in Y polarisation state on the reference route. This minimises the power interfering with the modulated signal in the Y (marked with blue) polarisation state. Fig. 2 gives an idea about the difference that this polariser makes. The upper plot of Fig. 2 is depicting the output waveform without polariser, while the lower one with it. It can be seen that the well placed polariser make the impulses much more symmetric, more square wave-like. The improvement is significant.
- We might achieve the same grade of polarisation of the reference signal with less explicit modification. If a PBS 2 is chosen to have high polarisation extinction ratio (PER), it will act better at suppressing the Y polarised contribution of the reference signal before routing to the common (XY) port. However, a commercially available polariser works with at least 20 dB of additional suppression in the undesired polarisation state, while increasing the PER of a normal PBS by 20 dB is rather unrealistic.

To conclude the above, the best method to further increase the separation of the modulated and reference signals is to make the reference signal more polarised before recombining it with the modulated one. The easiest and most straightforward way to this is to extend the setup with a X aligned polariser between the Splitter and PBS 2.

### *B. Pre-impulses*

After the first simulations we have noticed unexpected, small amplitude impulses before every normal (expected) impulse on the receiver side (also shown in Fig. 2). They appeared 500 ns before the expected impulses, which was the exact amount of delay caused by the optical delay line in the circulator-like structure. This allowed us to conclude, that these pre-impulses impulses are present, because a certain fraction of optical power bypasses this delay line, meaning that there is no full separation between the polarised individual ports (X and Y polarised) of the polarisation beam splitters. The principle of the operation requires both reference and modulated signal in the receiver to actually produce an output signal, otherwise we wouldn't get pre-impulses.
At first glance it would seem that the root of this impairment is a tiny fraction of modulated optical signal avoiding PBS 1 on Alice side and finding its way to the balanced receiver, while a small fraction of the reference signal is also bypassing PBS 4 in Bob's device. In this scenario pre-impulses are forming the same way as the useful, high amplitude impulses, but with bypassing the circulator-like structures on both sides. In this case we should be seeing a small power copy before all impulse, but this is not what we experienced. Our simulations showed that pre-impulses are always having the same polarity (regardless of the polarity of the subsequent useful impulse), and are only measured in one quadrature. With further simulations we have proven that PBS 1 has no effect whatsoever on the pre-impulses. Thus, the original assumption is false, the answer must be found on Bob's device.

The actual mechanism causing this issue is only the reference signal itself. The reference signal is exiting Alice's receiver with small attenuation in its way, consequently it will enter Bob's receiver with a relatively large power. After the polarisation controller it is routed to the phase modulator by PBS 3, but a small amount of X polarised reference signal will immediately get to the balanced receiver through the Y port of PBS 3 with no delay (due to imperfections of polarisation splitter defined by PER). At the same time most of its power passes the phase modulator and is routed to PBS 4, where in theory it should be delayed and have its polarisation modified. Because PBS 4 also has a certain grade of crosstalk (about -60 dB) between its X and Y ports, a small fraction of power is not delayed, but is going immediately in the balanced receiver. The proof of this concept is that the magnitude of pre-impulses

TABLE I
EFFECT OF PARAMETER MODIFICATION OF CERTAIN DEVICES EXPRESSED IN EVM

| | Modification | EVM | Change in EVM |
|---|---|---|---|
| 1 | N/A (reference) | 4.39 % | 0% |
| 2 | Alice polariser 15 dB ⇒ 25 dB | 3.30 % | 1.09 % |
| 3 | Alice polariser 15 dB ⇒ 35 dB | 3.17 % | 1.22 % |
| 4 | Alice PBS 1 PER 20 dB ⇒ 30 dB | 4.35 % | 0.04 % |
| 5 | Alice PBS 1 PER 20 dB ⇒ 40 dB | 4.34 % | 0.05 % |
| 6 | Alice PBS 2 PER 20 dB ⇒ 30 dB | 3.30 % | 1.09 % |
| 7 | Alice PBS 2 PER 20 dB ⇒ 40 dB | 3.16 % | 1.23 % |

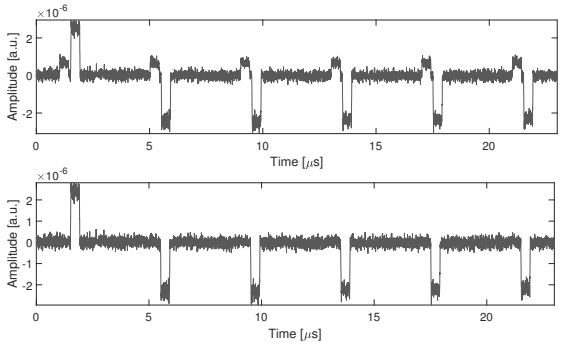



Fig. 3. Comparison of output waveforms without and with polariser on Bob's side

is independent of all devices on Alice's side, however the polarisation extinction ratios of the PBSs in Bob's device affect it: choosing a larger PER reduced the pre-impulse magnitude. Suggestions to address this problem:

- First we should choose PBSs for our setup, those have a larger grade of separation between their polarised ports (X and Y). This is rather hard to do, because manufacturers don't usually optimise this specific parameter, this is not even included in most of the PBS data sheets.
- We might also look for PBSs with larger PER. This is one of the most important parameters to look for in the data sheet, much more realistic, than optimising the crosstalk between X and Y ports.
- In case we don't have any further chance to swap the existing the PBSs on Bob' side, employing a polariser can also make a huge difference. The main cause of the problem (pre-impulses) is the X polarised reference signal routed in the same place with the Y polarised modulated signal, namely to the Y port of PBS 3. We might reduce this X polarised undesired power by using a Y aligned polariser between the PBS 3 and the balanced receiver. We are showing the effect of this polariser in Fig. 3. By comparing the two plots it might be seen that the pre-impulses ceased to exist on the lower one due to the polariser.

### C. Numerical results

In Table 1 we summarise the most important parameters of the polarisation dependent components affecting the transmission quality. We express the results in EVM. The first line is giving the EVM in the base parameter setting (reference), the other lines all contain one modification compared to this state. These changes basically all have the same physical effect, they make the light more polarised in their location. The most explicit way to do this is the use of polariser as seen in the second and third row of Table 1. By increasing the polarisation extinction ratio of the output combiner on Alice's side, we get the same result. This device polarises both the modulated and reference signals, therefore it is a bit more effective than the polariser, which only affects the reference signal. This becomes apparent if we compare row 3 with row 7. However, increasing the PER beyond a certain level is basically impossible due to manufacturing difficulties, so it is recommended to use a polariser to experience about the same result, at lower cost and effort. By improving the parameters of the PBS in the circulator like structure in the transmitter (PBS 1), less significant changes can be observed. This is because PBS 1 only deals with the modulated signal, while from separation point of view the more important one to handle is the reference signal. According to our simulations, all the other passive polarisation dependent components didn't have considerable effect on transmission quality.

## IV. POLARISATION CONTROLLING

Polarisation controlling has dedicated importance for proper operation. In our CVQKD system we are using a polarisation controller to correct the polarisation changing effect of the standard fibre connecting Alice and Bob. If it's not working properly, PBS 3 will not be able to correctly separate the modulated and reference signals resulting in an error. The CVQKD system at our university utilises a General Photonics POS–002 controller. It is working to maximise the power measured of a reference point chosen by us. The location of the controller is fixed (Bob's input), but we are free to choose the reference point. In this section we are dealing with the proper choice for reference point, moreover we are evaluating the effect of controlling error.

### A. Reference point

The root cause of most of the previously detailed impairments is the interference of the modulated and reference signals, mainly because of the unsatisfying behaviour of polarisation splitters and combiners (these two are physically identical). The polarisation controller is basically doing the same job, it is trying to minimise interference, therefore we must choose the location of the reference very carefully

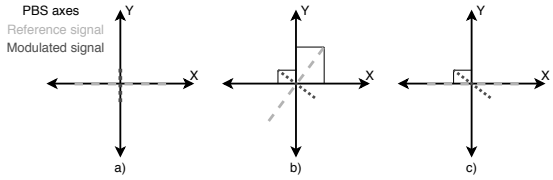



Fig. 4. Schematic operation of polarisation splitters

to address this issue. We also know - based on previously detailed considerations -, that the leakage of the reference signal into the routes of the modulated signal poses with the biggest threat. Thus the modulated and reference signals are not of equal importance, we have to pay more attention to the proper separation of the reference optical signal.

Fig. 4 is a simple depiction of how the polarisation splitter sets its X and Y (slow and fast axis in other words) outputs. Fig. 4.a depicts an ideal scenario, in which the reference signal (marked with red) is linearly polarised and its full power is matched with the X axis of the PBS. The same goes for the modulated signal (marked with blue), its power is in the orthogonal state of polarisation and is perfectly aligned with the Y axis. In this case the PBS is able to separate the two, leaving no interference: X output will only have the reference signal, Y output only the modulated one. In case Fig. 4.b the two signals are still orthogonal in polarisation, but there is a slight rotation compared to the axes of the PBS. This is the case of a certain degree of polarisation controlling error. The PBS is unable to separate the two, they are going to interfere. X and Y outputs will contain some of both the modulated and some of the reference signal, depending on the projection, as seen in Fig. 4. It is apparent, that we are looking to achieve scenario Fig. 4.a. We also must see, that the polarisation of the two signals will not always be orthogonal. Its because that they are processed in separate optical paths, travelling through different components with various properties, so they will not be orthogonal when recombined again. This is shown on Fig. 4.c. In case Fig. 4.b, if we use good polarisation controlling mechanisms, we are able to perfectly split the two signals, in case Fig. 4.c this will be impossible. This must be kept in mind when choosing the reference point for the controller, because Fig. 4.c is exactly what is happening in our system. We must be careful to keep concentrate the reference optical power to one output not to interfere with the modulated signal. Power leakage in the other way is accepted, due to the large differences in power. To draw the conclusion, the reference point must be at the X output of PBS 3, to concentrate the reference power in that path.

### *B. Controlling error*

In the Table 2 we are giving an overview of the effect of different magnitudes of polarisation controlling errors. We are starting from 0.001 degrees and moving with logarithmic steps to $1^\circ$. Controlling error will affect the efficiency of PBS 3 in separating the reference and modulated signals. This error cannot be corrected later in the system, therefore it is crucial achieving the best possible operation.

TABLE II
EFFECT OF ERROR IN POLARISATION CONTROLLING

| Error | EVM (%) | Difference (%) |
|---|---|---|
| $0^\circ$ | 3.4654 (reference) | 0 |
| $0.001^\circ$ | 3.4656 | 0.0002 |
| $0.01^\circ$ | 3.4930 | 0.0276 |
| $0.1^\circ$ | 4.9620 | 1.4966 |
| $1^\circ$ | 33.1339 | 29.6685 |

As seen on Table 1, $0.01^\circ$ error is not a significant error, but with $0.1^\circ$ the EVM is starting to rise dramatically. At $1^\circ$ the transmission is basically collapsed. $0.1^\circ$ error might be expressed in dBs: it resembles -55 dB of crosstalk between the reference and modulated signals. It might seem to be low, but as mentioned earlier, the system is very sensitive due to the large power differences.

## V. CONCLUSION

In this paper we used simulations to evaluate the performance of an optical transmission network suitable for continuous-variable quantum key distribution (CVQKD) to improve its performance. While most papers on this field focus on the issue of quantum theory, protocols and the construction of novel architectures, we gave practical considerations for the construction of a specific systems from a photonic engineer's point of view. We have proven that the crucial passive components in the system are the polarisation dependent devices, mainly the polarisation splitters and combiners (PBSs), whose insufficient behaviour reduce the separation of the modulated and reference signal impulses. We have shown that the proper parameter choice of PBSs is very important, and the transmission quality may be further improved using polariser. We have found the best possible location for extra polarisers in the system and validated our assumption in theory and also with simulations. This turned out to be a very straightforward modification, because a relatively simple and cheap passive device in the right place makes a large difference in terms of transmission quality. We also looked at the question of polarisation control, which is a defining factor for long-term stability. We simulated what magnitude of error might be accepted in the system and also found the ideal operation conditions for our specific controller. Our results might be able to give guidelines for the construction of CVQKD systems on the level of the optical backhaul network.

## ACKNOWLEDGMENT

The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications). The authors would like to special thank Dr. Zsolt Kis for the helpful suggestions.

## REFERENCES

[1] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. Proceedings *35th Annual Symposium on Foundations of Computer Science*. Santa Fe, NM, USA, 1994, pp. 124-134. DOI: 10.1109/SFCS.1994.365700

[2] Bennett, Charles & Zekrifa, Djabeur Mohamed Seifeddine. Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science - TCS*, 1984, vol. 560, p. 175-179.

[3] Marie, Adrien & Alleaume, Romain.. Self-coherent phase reference sharing for continuous-variable quantum key distribution. *Physical Review*, 2016, A. 95. DOI: 10.1103/PhysRevA.95.012316.

[4] Bisztray, Tamas & Bacsardi, Laszlo. (2018). The evolution of free-space quantum key distribution. Infocommunications Journal. 10. 22-30.

[5] Galambos, M. & Bacsardi, Laszlo. (2018). Comparing calculated and measured losses in a satellite-Earth quantum channel. Infocommunications Journal. 10. 14-19.

[6] Poppe, Andreas & Peev, Momtchil & Maurhart, Oliver. Outline of the SECOQC quantum-key-distribution network in Vienna. *International Journal of Quantum Information*, 2008, 6. 209-218. DOI: 10.1142/S0219749908003529.

[7] Peev, Momtchil et al., The SECOQC quantum key distribution network in Vienna. *New Journal of Physics*, 2009, 11. 075001. DOI: 10.1088/1367-2630/11/7/075001.

[8] Mraz, Albert et al., (2017). Quantum circuit-based modeling of continuous-variable quantum key distribution system: SIMULATION RESULTS OF A NOVEL CVQKD CIRCUIT. International Journal of Circuit Theory and Applications. DOI: 10.1002/cta.2347.

[9] P. Jouguet, S. Kunz-Jacques, A. Leverrier, P. Grangier And E. DIiamanti. Experimental demonstration of continuous-variable quantum key distribution over 80 km of standard telecom fiber. *CLEO: 2013*, San Jose, CA, 2013, pp. 1-2.

[10] N. Hosseinidehaj, Z. Babar, R. Malaney, S. X. Ng AND L. Hanzo. Satellite-Based Continuous-Variable Quantum Communications: Stateof-the-Art and a Predictive Outlook. *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 881-919, Firstquarter 2019. DOI: 10.1109/COMST.2018.2864557

[11] Z. Qu And I. B. Djordjevic. RF-assisted coherent detection based continuous variable (CV) QKD with high secure key rates over atmospheric turbulence channels. *2017 19th International Conference on Transparent Optical Networks (ICTON)*, Girona, 2017, pp. 1-5. DOI: 10.1109/ICTON.2017.8024919

[12] Xiaoxiong, Zhang & Zhang G, Yi-Chen & Li, Zhengyu & Yu, Song & Guo, Hong. 1.2 GHz Balanced Homodyne Detector for Continuous-Variable Quantum Information Technology. *IEEE Photonics Journal, 2018*, PP. 1-1. DOI: 10.1109/JPHOT.2018.2866514.

[13] X. Wang, J. Liu, X. Li And Y. Li. Generation of Stable and High Extinction Ratio Light Pulses for Continuous Variable Quantum Key Distribution. *IEEE Journal of Quantum Electronics*, June 2015, vol. 51, no. 6, pp. 1-6, Art no. 5200206. DOI: 10.1109/JQE.2015.2427031

[14] P. Kucera, Quantum Description of Optical Devices Used in Interferometry. *Radioengineering*, 2007, vol. 16, no. 3.

[15] Sibson, Philip & Kennard, Jake & Stanisic, Stasja & Erven, Chris & O'Brien, Jeremy & G Thompson, Mark. ntegrated Silicon Photonics for High-Speed Quantum Key Distribution. *Optica*, 2016, 4. DOI: 10.1364/OPTICA.4.000172.

**David Kobor** received his B.Sc. from the Budapest University of Technology and Economics (BME), in 2018. He is a member of the Balatonfüred Student Research Group. He is currently pursuing a Master's degree at the same institute specialised in Wireless Networks and Applications. His professional interests include simulation and measurement of optical systems, such as Radio over fibre applications and optoelectronic oscillators.

**Eszter Udvary** received a Ph.D. degree in electrical engineering from Budapest University of Technology and Economics (BME), Budapest, Hungary, in 2009. She is currently an Associate Professor at BME, Department of Broadband Infocommunications and Electromagnetic Theory, where she leads the Optical and Microwave Telecommunication Lab. Dr. Udvarys research interests are in the broad areas of optical communications, include optical and microwave communication systems, Radio over fibre systems, optical and microwave interactions and applications of special electro-optical devices.

# GrAMeFFSI: Graph Analysis Based Message Format and Field Semantics Inference For Binary Protocols, Using Recorded Network Traffic

Gergő Ládi[1], Levente Buttyán[2], and Tamás Holczer[3]

***Abstract*—Protocol specifications describe the interaction between different entities by defining message formats and message processing rules. Having access to such protocol specifications is highly desirable for many tasks, including the analysis of botnets, building honeypots, defining network intrusion detection rules, and fuzz testing protocol implementations. Unfortunately, many protocols of interest are proprietary, and their specifications are not publicly available. Protocol reverse engineering is an approach to reconstruct the specifications of such closed protocols. Protocol reverse engineering can be tedious work if done manually, so prior research focused on automating the reverse engineering process as much as possible. Some approaches rely on access to the protocol implementation, but in many cases, the protocol implementation itself is not available or its license does not permit its use for reverse engineering purposes. Hence, in this paper, we focus on reverse engineering protocol specifications relying solely on recorded network traffic. More specifically, we propose GrAMeFFSI, a method based on graph analysis that can infer protocol message formats as well as certain field semantics for binary protocols from network traces. We demonstrate the usability of our approach by running it on packet captures of two known protocols, Modbus and MQTT, then comparing the inferred specifications to the official specifications of these protocols.**

***Index Terms*—protocol reverse engineering, message format, field semantics, inference, binary protocols, network traffic, graph analysis, Modbus, MQTT**

## I. INTRODUCTION

Protocols describe the formats, types, contents, and sequence of messages that are sent and received in order to exchange data between the communicating parties, as well as the rules according to which these messages must be processed. The protocols themselves are defined in specifications, which are not always available to the general public. This is unfortunate, as having access to specifications is required for the generation of models that serve as the basis of several security-related applications, such as the development of intrusion detection systems (IDS) that understand the protocol and can raise alarms when anomalous protocol messages are detected [1], the creation of protocol-specific honeypots that simulate a device running said protocol for attacker behaviour analysis [2], and fuzz testing protocol implementations for programming errors or hidden features [3].

Protocol reverse engineering is an area of study that provides methods which aim to reconstruct the specifications for protocols where these are not available. Given that manual reverse engineering of protocols is rather time consuming, and that new protocols appear frequently, it is generally recommended that an automated approach be used. These aim to provide at least partial information about protocols in at least a semi-automated fashion, typically relying on the analysis of captured network packets or existing protocol implementations (binaries), or a combination of these [4]. However, protocol implementations may not always be available, and licensing restrictions or user agreements may forbid such reverse engineering. For this reason, we focus on methods that only rely on captured network traffic.

The reverse engineering process is usually comprised of three main phases [5]. The first phase involves setting up the environment in which the analysis will be conducted, as well as performing the necessary preparation steps such as generating and capturing network traffic. The second phase focuses on determining the types of the possible messages (i.e. messages that result in functionally distinct behaviour from the other party) along with the semantics of the fields (groups of bytes) within the messages. The third phase focuses on constructing a state machine for the protocol, which describes the valid sequences of the previously determined message types (i.e. the grammar of the protocol), however, we do not aim to reconstruct the state machine in this paper.

To measure the goodness of the inferred specifications, typically three metrics are used: correctness, conciseness, and coverage [4], where correctness measures what percentage of the inferred messages represent true messages, conciseness shows how many inferred messages represent one true message, and coverage shows what portion of the true message types were found.

Based on how messages are represented, protocols can be classified into two groups: plain text and binary. Plain text protocols such as Hypertext Transfer Protocol (HTTP) or Simple Mail Transfer Protocol (SMTP) exchange human-

[1,2,3] Laboratory of Cryptography and System Security, Department of Networked Systems and Services, Budapest University of Technology and Economics, Budapest, Hungary
[1] BME Balatonfüred Student Research Group, Hungary
E-mail: {gergo.ladi, buttyan, holczer}@crysys.hu

DOI: 10.36244/ICJ.2020.2.4

readable messages where the fields are separated by delimiters such as spaces, colons, or new line characters, and at least one field contains a keyword that determines how the message should be interpreted. On the other hand, binary protocols such as Server Message Block (SMB) or Modbus exchange binary messages that are not human-readable, lack field separators, and one or more groups of bytes determine how the message should be interpreted.

In this paper, we present GrAMeFFSI, a novel graph analysis based algorithm for binary protocols which can infer not only the message types, but also a variety of field semantics, using only network traces of the protocols. We implement and test the algorithm on real-world captures of two commonly used binary protocols, Modbus and MQTT, achieving perfect correctness and completeness scores as well as decent conciseness scores that surpass those of existing state-of-the-art methods. In addition, we introduce two metrics, accuracy and adjusted accuracy, to measure the goodness of semantics inference. We also show that GrAMeFFSI can infer field semantics with over 95% accuracy if high quality network traces are available.

This paper revises, improves, and extends our previous work, *Message Format and Field Semantics Inference for Binary Protocols Using Recorded Network Traffic* [6]. Notable additions are a model merging phase in the algorithm and the mathematical formalization of the metrics. The model merging phase further improves the accuracy of our algorithm while also providing extra semantical information, and the formalization aims to make our results possible to reproduce as well as make it easier to compare it to other works (where such metrics are used).

The rest of the paper is structured as follows: in Section II, we discuss related work. In Section III, we present our algorithm in detail, along with additional possible optimization steps. Next, in Section IV, we evaluate the previously presented algorithm on packet captures of two common protocols, Modbus and MQTT. Then, in Section V, we briefly discuss the possible limitations of our solution, followed by opportunities for future work. Finally, Section VI concludes our paper.

## II. Related Work

Protocol reverse engineering dates back to the 1950s, where it typically meant the analysis of finite state machines for fault detection [7]. The first well-known project that aimed at restoring the specifications of a computer protocol was the Protocol Informatics Project by M. A. Beddoe [8] in 2004, which used bioinformatical algorithms such as the well-known Needleman-Wunsch sequence alignment algorithm on network traces to infer the message types of the text-based protocol HTTP. It was later followed by Discoverer [9], Biprominer [10], ReverX [11], ProDecoder [12], and AutoReEngine [13] that all relied only on network traffic. While most algorithms aimed at reversing both text-based and binary protocols, some specialized in one or the other, typically achieving better performance metrics compared to the more general solutions of their time. Biprominer, as its name suggests, targeted binary protocols, while ReverX targeted text-based protocols. The methods employed vary – Discoverer relies on sequence alignment, Biprominer and AutoReEngine leverage data mining approaches, while ProDecoder makes use of natural language processing algorithms.

Early works typically focused on reverse engineering the message formats and their syntax, and did not put much emphasis on inferring field semantics (that is, what each of the fields means). Even those that tried did not achieve significant results – Discoverer admits to achieving between 30-40% accuracy [9], and not even Netzob exceeds 50% [14]. FieldHunter [15] from 2015 was the first to achieve over 80% accuracy on semantics.

Methods relying on reversing implementations appeared under the names of Polyglot [16], AutoFormat [17], and ReFormat [18]. These generally work on the principles of dynamic taint analysis, marking pieces of code in the memory area of a running executable that are run in response to a given message, then making assumptions about the message formats based on what and how was marked. It has been proven [4] that binary analysis based approaches can achieve better results, however, purely traffic analysis based approaches are also important as binaries may not always be at our disposal and legal agreements may prevent us from analysing or reverse engineering these.

Solutions to reverse the protocol grammar (the state machine of the protocol) have also been proposed in the form of ScriptGen [19], Prospex [20], Veritas [21], and MACE [22]. However, they are not in scope of this paper as we currently do not aim to reconstruct the state machine of the protocol.

In this paper, we aim to compete with Discoverer, Biprominer, and ProDecoder, three different approaches for reversing the message formats of binary protocols; as well as Netzob and FieldHunter that aim at extracting semantic information. The performance statistics of these solutions, as given by their authors (or calculated based on their respective papers), are shown in Table I.

We believe that no prior protocol message format reversing method exists that is based on graph operations.

## III. Our Approach

Our approach consists of five distinguishable phases. The first phase is a preparation phase, in which data is gathered and transformed such that it can be processed in the second phase. The second phase is the core algorithm that constructs directed acyclic connected graphs (rooted trees) based on the input. Next, in the third phase, we merge the trees from phase two, following a set of rules. In the fourth phase, (optional) optimizations may be run on the trees. These optimizations generally improve a certain metric at a possible cost of impairing a different metric. Finally, the resulting tree is used to enumerate the inferred message types and field semantics.

TABLE I
PERFORMANCE METRICS OF SIMILAR APPROACHES

| Approach | Correctness | Conciseness | Coverage | Accuracy | Tested on # protocols |
|---|---|---|---|---|---|
| Discoverer | 0.9 | 5 | 0.95 | 30-40% | 3 |
| Biprominer | 0.99 | Unknown | 0.967 | N/A | 3 |
| ProDecoder | 0.975 | Unknown | 0.975 | N/A | 2 |
| Netzob | 0.775 | 1.74 | Unknown | 33.4% | 4 |
| FieldHunter | Unknown | 2.1 | Unknown | 91.89% | 7 |

Notes: Values for Netzob are only approximately accurate as they were manually read from a plot. For FieldHunter, only binary protocols were considered.

### A. Preparations

In the preparation phase, the environment needs to be planned and set up. In order to observe and record protocol traffic, at least one client and at least one server application instance (or in the case of peer-to-peer applications, two instances) should be running. These instances may or may not be running on the same device, and if multiple devices are used, these need not be of the same type (e.g. one can be an ordinary computer, while the other an industrial programmable logic controller (PLC)). This approach needs no access to the source code or the compiled application binaries, nor does it need access to the memory of the devices where these are running. The only requirement is that there has to be a way to monitor and capture network traffic flowing between the application instances. This is typically done by running *tcpdump* or *Wireshark* on one of the devices or connecting them via a hub (or a switch with port mirroring configured), and then capturing traffic from a third device that is also connected to the hub.

Once the environment is set up and the capture is running, traffic should be generated by invoking as many features of the client with as many different options and in as many different combinations as possible, all repeated a number of times. This ensures that most of the message space is covered, which is essential for near-complete and accurate recovery of the protocol specification.

It is highly preferable to repeat the traffic generation procedure a couple of times, disconnecting and reconnecting the client and the server (or the peers) in between. This ensures that multiple flows (sessions, connections) are recorded. Since certain values such as session identifiers never change during a single session, recording multiples of them is necessary in order to achieve more accurate results. Similarly, if multiple clients and servers (or peers) are available, it is also imperative to record at least one full session in each possible valid combination thereof. This ensures that fields containing identifiers that are unique and never change for each client (e.g. factory-set device IDs) can still be detected as such.

### B. Tree Construction

In the second phase, the recorded traffic is processed and a tree is constructed for each flow based on the messages that appear in that given flow. These trees represent the suspected message types and field semantics as deduced from the data seen, and will be further processed in later steps.

Each captured packet is read into the memory. For each packet, a pointer is assigned that initially points to the first byte of the packet. This pointer is later used to keep track of how many bytes have already been processed in that specific packet. A separate pointer is needed for each packet as some steps of the algorithm increment this pointer by different amounts for different packets.

The algorithm maintains and builds a graph that initially consists of one node, the root node (which also is a leaf at this point). In each step, new nodes of different colours are appended to one of previous leaves. The colours are used to indicate the inferred field semantics, and are based on the following decisions:

1) Constants - Check the next byte of each packet. If this is the same for all packets, consider this byte a constant. Append a green leaf to the current branch, advance all pointers by one, then continue processing at 1).
2) Length-prefixed strings - Interpret the next byte as an integer, then test whether this value is followed by this many printable characters. If this test succeeds, a length-prefixed string was found. Append a cyan leaf to the current branch, advance all pointers by one plus the length of the string, then continue processing at 1).
3) Null-terminated strings - Starting from the next byte in each packet, test whether the following bytes can be interpreted as a sequence of printable characters followed by a null byte. If this test succeeds, a null-terminated string was found. Append a cyan leaf to the current branch, advance all pointers past the next null byte, then continue processing at 1).
4) Length fields - Interpret the next four bytes in each packet as a single integer. Test whether this value matches the length of packet (optionally with a given offset). If the test succeeds, these four bytes indicate the length of the packet. Append a blue leaf to the current branch, advance all pointers by four, then continue processing at 1). If the test fails, repeat the same procedure but with the next two bytes only instead of four. If that fails as well, repeat the procedure, this time just with the next single byte.
5) Counters - Interpret the next four bytes in each packet as a single integer. Test whether this value increases by the same amount between packets. If the test succeeds, these four bytes form a counter. Append a purple leaf to the current branch, advance all pointers by four, then

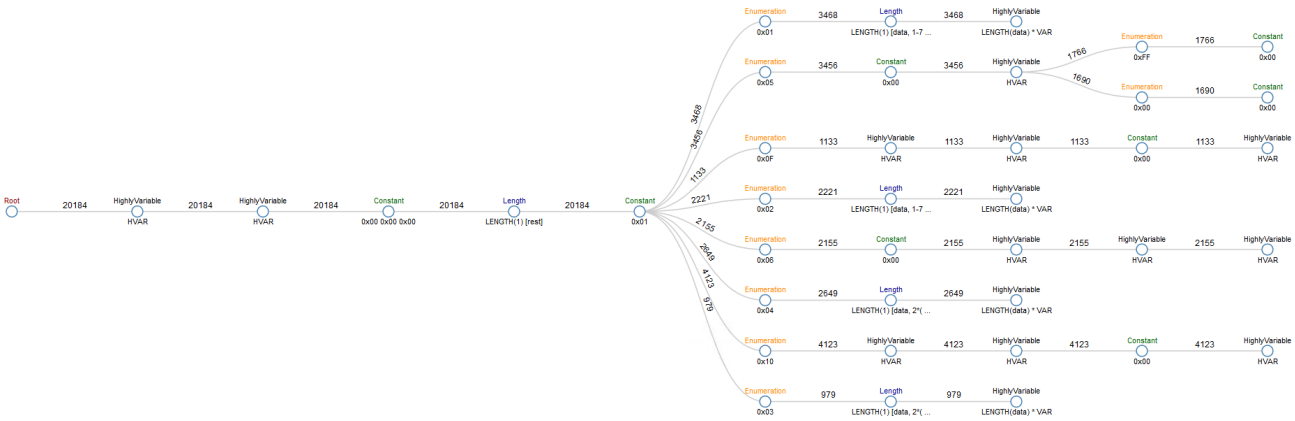



Figure 1. Output of the tree builder algorithm showing the results of a run on a capture of responses of the Modbus protocol.

continue processing at 1). If the test fails, repeat the same procedure but with the next two bytes only instead of four. If that fails as well, repeat the procedure, this time just with the next single byte.

6) Enumerated types - Check the next byte of each packet. Calculate how many distinct values occur. If this amount is lower than a threshold, we have found an enumerated type. For each distinct value that was seen, append an orange leaf to the current branch, and tag it with one of the previously unused distinct values. Split the list of packets such that each packet is assigned to the branch that is tagged with the value of the packet's next byte. From this point on, only process messages that were assigned to the branch that is currently being processed. Advance all pointers by one. Continue processing at 1) for each of the newly created branches. Since branches are not interdependent, if multiple CPU cores are available, processing may continue in parallel. As for the threshold, based on empirical evidence, values between 8 and 20 seem to be ideal, or if the number of distinct message types is suspected, that number should be used instead.
7) Highly variable - If none of the previous classifiers classified this byte as something else, then it takes on many different values that follow no discernible pattern. Append a black leaf to the current branch, advance all pointers by one, then continue processing at 1).

When no packet on any of the branches has unprocessed bytes left, no more nodes can be added to the tree, and the algorithm ends, outputting the tree. An example of a result can be seen on Figure 1. Note that the colours of the nodes may be arbitrarily chosen as long as each field type is coloured differently.

### C. Model Merging

If we just considered each flow individually, it would not be possible to find mutually exclusive message types (as at least one of these would be missing in each flow), and it would also not be possible to find fields containing session identifiers (as these would appear constant within each flow). However, merging the trees and correlating data from the previous step solves such issues, greatly improving the resulting inferred specification if the right network traces are available.

The merging process is as follows: starting from the root node, compare the next child node of each tree using the following rules:

1) If all are of type *counter*, *flag*, *length*, *string* or *variable*, continue merging the direct descendants.
2) If all are of type *enumerated*, continue merging each subtree where the value of the *enumerated* node is the same (this may be parallelized). If a value only appears once, add it with all of its children to the resulting tree. Alternatively, if this step results in too many (exact value varies on a case-by-case basis) branches, this may be the case where a *variable* type gets detected as an *enumerated type* due to the inputs being poor – in this case, the *enumerated type* may be replaced with a *variable* type, and all subtrees may be merged into one.
3) If all are of type *constant*, check the value of the nodes. If the value is always the same, it's a generic constant. If the value is always the same for the same client (or server) and is different for other clients (or servers), it's a source or destination host identifier. If the value is only the same within each flow, then it must be a session identifier.
4) If some are of type *constant* and all others are of the same (non-constant) type, proceed as if everything was of that other non-constant type.
   For example, suppose we have five trees. The next element is a *length* field according to three of these, and it's a *constant* according to the other two. The field should be treated as if it was a *length* field in all five trees.
5) Any other combinations are rare and typically indicate a problem with the recorded traffic or the implementations themselves. In such cases, the field should be treated as if it was of type *variable* in all of the trees.

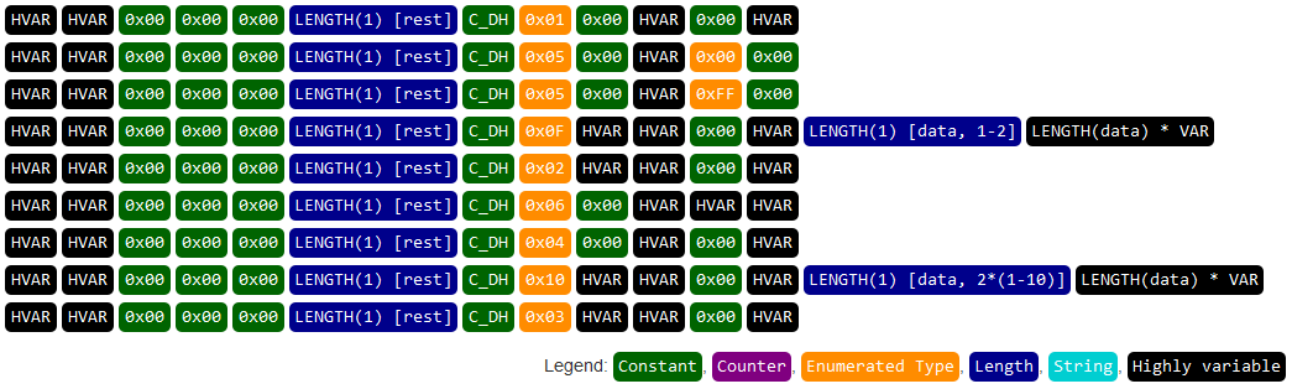



Figure 2. Message types of Modbus requests, as read from a graph. Each line represents a unique (detected) message type, with each block denoting a group of bytes (coloured as per the legend). For constants and enumerated types, their values are displayed in the blocks. For length and counter types, their widths and seen value ranges are shown. For everything else, the type of the node is displayed.

For example, an implementation might generate request identifiers sequentially, while others might choose them randomly. In this case, the field containing the request identifier will be recognized as a *counter* for the former implementations, while it will be recognized as *variable* for the latter ones.

### D. Optimizations

Assuming that the protocol being analysed only consists of messages that only contain fields of the previously listed detectable properties, and that the input is of high enough quality (i.e. there are enough messages to analyse on each branch), the tree construction algorithm yields a correct but not necessarily concise result. The resulting tree may be further optimized for one or more metrics, usually at a cost of others.

- Variable length messages - Certain message types, such as write requests with payloads of varying length or responses to read requests will get inferred multiple times: once for each different message length. This phenomenon may be detected by looking for branches that end in a number of highly variable fields that are preceded (not necessarily directly) by a length byte, and are otherwise identical. Message types detected this way may be merged to improve the conciseness score.
- Falsely detected enumerated types - Protocols may contain bytes that contain fields that have a limited range of values (e.g. flags) but don't change the rest of the message structure. These will be inferred as enumerated types, possibly resulting in the same message type(s) getting recognized multiple times. This phenomenon may be detected by looking for identical branches that are preceded by the enumerated type in question. In this case, the branches may be merged and the enumerated type node may be replaced by a brown coloured (Flag) node. This may improve the conciseness score, but may also incorrectly merge truly different message types, resulting in loss of correctness.

### E. Interpreting the Results

Once the tree construction is done, and the optional optimization steps are run, the distinct message types may be read from the graph by considering the walks from the root to each leaf node. An example of results can be seen on Figure 2:

## IV. Evaluation

The goodness of message type inference was measured by the three standard metrics, correctness (1), conciseness (2), and coverage (3):

$$Correctness = \frac{|I \cap T|}{|I|} \tag{1}$$

$$Conciseness = \frac{|I| - |I \setminus T|}{|T| - |T \setminus I|} \tag{2}$$

$$Coverage = \frac{|T \cap I|}{|T|} \tag{3}$$

where $T$ is the set of true messages and $I$ is the set of inferred messages.

To calculate these three metrics, we need the true and the inferred models of the message types, as well as a network capture that contains each true message type at least once. Then, the following algorithm can be used:

1) Initialization: Begin with an empty list of mappings, $M_{TI}$, which will contain mappings from true message types to inferred message types.
2) Mapping creation: For each protocol message that exists in the network capture: find out which message type it corresponds to in the sets of true and inferred message types. If it matched something in both sets, say, $T_x$ among the true message types and $I_y$ among the inferred message types, then add a $T_x \mapsto I_y$ mapping to $M_{TI}$. (If the exact same mapping is already on the list, it should not be added a second time.)

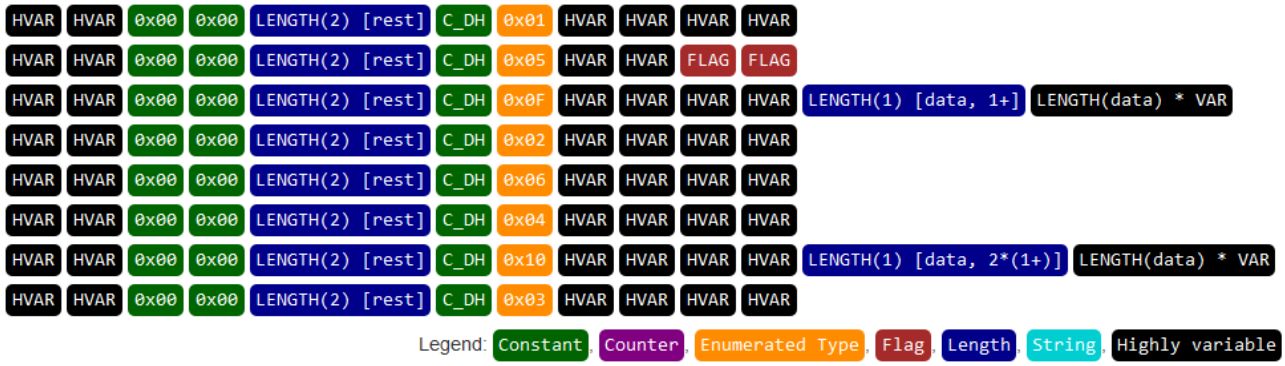



Figure 3. A model of Modbus requests, built based on the true specification. To be interpreted in the same way as Figure 2.

3) Correctness: count the number of distinct $T_i$s that appear on the left-hand side in mappings in $M_{TI}$ – this is the number of correctly inferred message types. Count the number of $I_j$s that never appear on the right-hand side in mappings in $M_{TI}$ – this is the number of bogus (inferred but nonexistent) messages types. Finally, to get the correctness, divide the number of correctly inferred types by the sum of correctly inferred and bogus message types.
4) Conciseness: subtract the number of bogus message types from the total number of inferred message types. Divide this number by the number of true message types minus the number of message types that were not found. The number of message types that were not found can be calculated by counting the number of $T_i$s that never appear on the left-hand side in mappings in $M_{TI}$.
5) Coverage: Divide the number of correctly inferred message types by the number of elements in $T$.

To measure the accuracy of semantics inference, we defined two metrics: accuracy and adjusted accuracy. Accuracy measures what percentage of field semantics were inferred correctly, while adjusted accuracy accepts miscategorized bytes as correct where the miscategorization was a result of the input not being rich enough. For example, consider a two-byte counter that was classified as a one-byte constant followed by a one-byte counter. The accuracy metric considers this incorrect, since this does not strictly match the specification. However, it is considered correct for the adjusted accuracy metric, since this miscategorization was the result of the upper byte never changing values (thus the input not being rich enough).

To compute the accuracy and adjusted accuracy scores, we use a Tree Edit Distance (TED) algorithm. The TED is a measure of how similar two trees are. It is generally defined as the minimum cost sequence of edit operations that transforms one tree into the other (4) [23].

$$TED(t_1, t_2) = \min_{(e_1,\ldots,e_k)\in\mathcal{P}(t_1,t_2)} \sum_{i=1}^{k} c(e_i) \qquad (4)$$

We have chosen APTED [24, 25], one of the state-of-the-art TED algorithms. It supports three types of edit costs (weights): node insertion, node deletion, and node renaming (relabeling). It has a Java-based implementation available[4] on Github, which we ported to C# and published[5] on Github. Running APTED on the graphs of the true and the inferred specifications with the weights $(0, 0, 1)$ for insertion, deletion, and relabeling respectively, we get the number of bytes that were incorrectly inferred semantically. Subtracting this number from the total number of bytes in the graph, then dividing the result by the total yields the accuracy. Using $0$ as weights for insertion and deletion ensures that bogus and duplicate messages, as well as ones that were not found are not considered when calculating the accuracy of semantics inference. Adjusted accuracy is calculated similarly, by using $(0, 0, f(n_1, n_2))$ as weights, where $f$ returns $0$ not just when the labels of $n_1$ and $n_2$ are equal, but also when the inferred node is constant; in any other cases, $f$ returns $1$.

GrAMeFFSI was evaluated on two commonly used binary protocols, Modbus and MQTT.

### A. *Evaluation with Modbus Traffic*

Modbus is a communication protocol originally designed in 1979 for use with PLCs. Today, it is still frequently used with industrial control systems (ICS). Modbus' specification is openly available. Although the specification [26] defines 21 functions (pairs of requests and responses), some of these are only to be implemented for use over serial lines, and a typical implementation only contains 8 of these: 4 kinds of reads and 4 kinds of writes.

For the evaluation, we have recorded approximately 20 000 Modbus request-response pairs on an ICS testbed. This includes Modbus traffic from normal operation as well as several thousands of repeated manual read and write requests with a wide variety of legal parameter values. The source ports of the requests and the destination ports of the responses were edited to be the same with *editcap*, one of the tools from the Wireshark package. This editing was needed to make sure that the packets are recognized to belong to the same message flow. The Modbus payloads were not altered in any manner, nor were the IP addresses that are used to determine which device is which for host identifier inference.

[4] https://github.com/DatabaseGroup/apted
[5] https://github.com/GergoLadi/APTEDSharp/

TABLE II
PERFORMANCE METRICS OF THE ALGORITHM ON THE MODBUS PROTOCOL

| Algorithm | Message Type | Correctness | Conciseness | Coverage | Accuracy | Adjusted Accuracy |
|---|---|---|---|---|---|---|
| Tree construction with no optimizations | Request | 1 | 2.375 | 1 | 0.8 | 0.99 |
| Tree construction with optimization #1 | Request | 1 | 1.125 | 1 | 0.8 | 0.99 |
| Tree construction with optimizations #1 and #2 | Request | 1 | 1 | 1 | 0.81 | 1 |
| Tree construction with no optimizations | Response | 1 | 4.875 | 1 | 0.8409 | 0.9886 |
| Tree construction with optimization #1 | Response | 1 | 1.125 | 1 | 0.8409 | 0.9886 |
| Tree construction with optimizations #1 and #2 | Response | 1 | 1 | 1 | 0.8523 | 1 |
| Tree construction with no optimizations | Average | 1 | 3.625 | 1 | 0.8205 | 0.9893 |
| Tree construction with optimization #1 | Average | 1 | 1.125 | 1 | 0.8205 | 0.9893 |
| Tree construction with optimizations #1 and #2 | Average | 1 | 1 | 1 | 0.8312 | 1 |

Next, we built models of the Modbus requests and responses based on the true specification. An example of a model is shown on Figure 3). These were then used to calculate the performance metrics for the algorithm (see Table II for results). It can be seen that the algorithm reached maximum correctness and coverage, no matter what optimizations were enabled. Enabling both optimizations also maximized conciseness. The differences between accuracy and adjusted accuracy can be explained by the top bytes of length fields and highly variable fields getting detected as constants due to the input packet dump not being of high enough quality.

### B. Evaluation with MQTT Traffic

MQTT, or Message Queueing Telemetry Transport is a standard messaging protocol that follows the publish-subscribe pattern. MQTT is fully open, and is typically used in Internet-of-Things (IoT) solutions. The specification defines a total of 14 message types, 5 of which may only be sent by the client, 4 of which may only be sent by the server, and 5 of which may be sent by either party [27].

For the evaluation, we set up an environment with *Eclipse Mosquitto*[5], an open source MQTT server, then used the *HiveMQ Websocket Client*[6] to perform as many operations and with as many different parameter combinations as possible. Traffic was captured on the server using Wireshark, resulting in approximately 1 200 packets. The packets did not need to be altered in any way before analysis.

As with Modbus, we built models based on the true specification, to which we then compared our inferred specification. Results are shown in Table III. Perfect correctness and coverage are achieved in addition to decent conciseness. In the majority of cases, the low (unadjusted) accuracy scores can be attributed to the fact that several messages of the protocol are of fixed length, which results in GrAMeFFSI misclassifying length fields as constants.

## V. LIMITATIONS AND FUTURE WORK

During evaluation, we have found that the solution presented herein has two limitations that may not be possible to overcome:

- Handling encrypted traffic - Like any other approach that relies on nothing else but network traces, reconstruction fails if the protocol messages are encrypted or are otherwise obfuscated. If the encryption is weak or badly implemented, it may be cracked, or a man-in-the-middle attack may be used against the communicating parties. Failing that, a binary analysis based (or hybrid) approach may still work.
- Poor results for poor inputs - If certain message types were not seen during the capture process, those will be missing from the reconstructed specification, resulting in suboptimal coverage metrics. In addition, if messages for a given type were low in count or variance, then field semantics inference may fail, resulting in low accuracy scores.

We have also identified areas where GrAMeFFSI could be further improved:

- Detection of unicode strings - Currently, only ASCII strings can be detected, but newer protocols may contain messages having unicode strings. We expect that it is possible to detect these strings, however, extensive testing is needed to ensure that this functionality does not introduce false detections.
- Split-byte fields - Some protocols, including MQTT, don't always use whole bytes to store information (e.g. the upper four bits of a byte might be flags, while the lower four could be a counter). The algorithm could be reworked to try to detect and handle these cases.
- Leaving room for error - It is currently assumed that no packets are lost, duplicated or corrupted during transmission and capture. One of these events occurring may result in most types not being detected correctly. This issue could be worked around by allowing a small amount of corrupted or out-of-sequence packets. However, this could also result in false detections, thus should be a subject of further research.

With these improvements done, it would be possible to generate protocol specifications that are accurate enough to be used directly as a basis of fuzz testing, honeypots or firewall rules, among others. Furthermore, we plan to investigate how the results of the tree building algorithm could be used as inputs to other algorithms that aim to infer protocol grammar or otherwise try to find correlations between fields in requests and responses.

[6] https://projects.eclipse.org/projects/technology.mosquitto
[7] http://www.hivemq.com/demos/websocket-client/

TABLE III
PERFORMANCE METRICS OF THE ALGORITHM ON THE MQTT PROTOCOL

| Algorithm | Message Type | Correctness | Conciseness | Coverage | Accuracy | Adjusted Accuracy |
|---|---|---|---|---|---|---|
| Tree construction (any optimization settings) | Client | 1 | 1.2 | 1 | 0.5483 | 0.9677 |
| Tree construction without optimization #2 | Server | 1 | 1 | 1 | 0.7333 | 1 |
| Tree construction with optimization #2 | Server | 1 | 1 | 1 | 0.8 | 1 |
| Tree construction without optimization #2 | Shared | 1 | 2 | 1 | 0.7391 | 0.9565 |
| Tree construction with optimization #2 | Shared | 1 | 1 | 1 | 0.7391 | 0.9565 |
| Tree construction without optimization #2 | Average | 1 | 1.4 | 1 | 0.6735 | 0.9747 |
| Tree construction with optimization #2 | Average | 1 | 1.06 | 1 | 0.6958 | 0.9747 |

## VI. CONCLUSION

In this paper, we have presented GrAMeFFSI, a novel method to infer message types and field semantics for binary protocols. Our method relies exclusively on network traces, and works by constructing, merging, and optimizing acyclic graphs based on the contents of the packets in the trace. We have presented a methodology to evaluate the performance of the algorithm, then performed evaluations against the known specifications of two commonly used protocols. Based on the results, we conclude that the approach surpasses existing similar solutions in terms of correctness, conciseness and coverage, while also providing more accurate field semantics in most of the cases.

## ACKNOWLEDGMENT

The research presented in this paper has been partially supported by the Hungarian National Research, Development and Innovation Fund (NKFIH, project no. 2017-1.3.1-VKE-2017-00029), and by the IAEA (CRP-J02008, contract no. 20629). The first author has also been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications).

## REFERENCES

[1] H. J. Wang, C. Guo, D. R. Simon, and A. Zugenmaier, "Shield: Vulnerability-driven network filters for preventing known vulnerability exploits," Proceedings of the *ACM SIGCOMM 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pp. 193–204, 2004. DOI: 10.1145/1015467.1015489

[2] T. Krueger, H. Gascon, N. Krämer, and K. Rieck, "Learning stateful models for network honeypots," *Proceedings of the 5th ACM Workshop on Artificial Intelligence and Security*, pp. 37–48, 2012. DOI: 10.1145/2381896.2381904

[3] J. Antunes, N. Neves, M. Correia, P. Verissimo, and R. Neves, "Vulnerability discovery with attack injection," *IEEE Transactions on Software Engineering*, vol. 36, no. 3, pp. 357–370, 2010. DOI: 10.1109/TSE.2009.91

[4] J. Narayan, S. K. Shukla, and T. C. Clancy, "A survey of automatic protocol reverse engineering tools," *ACM Computing Surveys*, vol. 48, no. 3, 2016. DOI: 10.1145/2840724

[5] J. Duchêne, C. L. Guernic, E. Alata, V. Nicomette, and M. Kaâniche, "State of the art of network protocol reverse engineering tools," *Journal of Computer Virology and Hacking Techniques*, vol. 14, no. 1, pp. 53–68, 2018. DOI: 10.1007/s11416-016-0289-8

[6] G. Ládi, L. Buttyán, and T. Holczer, "Message format and field semantics inference for binary protocols using recorded network traffic," *26th International Conference on Software, Telecommunications and Computer Networks*, 2018. DOI: 10.23919/SOFTCOM.2018.8555813

[7] D. Lee and M. Yannakakis, "Principles and methods of testing finite state machines – A survey," *Proceedings of the IEEE*, vol. 84, no. 8, pp. 1090–1123, 1996. DOI: 10.1109/5.533956

[8] M. A. Beddoe, "Network protocol analysis using bioinformatics algorithms," http://www.4tphi.net/32awalters/PI/PI.html, 2004.

[9] W. Cui, J. Kannan, and H. J. Wang, "Discoverer: Automatic protocol reverse engineering from network traces," *SS'07 Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium,* 2007.

[10] Y. Wang, X. Li, J. Meng, Y. Zhao, Z. Zhang, and L. Guo, "Biprominer: Automatic mining of binary protocol features," *12th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, pp. 179–184, 2011. DOI: 10.1109/PDCAT.2011.25

[11] J. Antunes, N. Ferreira, and P. Verissimo, "ReverX: Reverse engineering of protocols," *12th International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT)*, 2011.

[12] Y. Wang, X. Yun, M. Z. Shafiq, L. Wang, A. X. Liu et al., "A semantics aware approach to automated reverse engineering unknown protocols," *20th IEEE International Conference on Network Protocols (ICNP)*, pp. 1–10, 2012. DOI: 10.1109/ICNP.2012.6459963

[13] J.-Z. Luo and S.-Z. Yu, "Position-based automatic reverse engineering of network protocols," *Journal of Network and Computer Applications*, vol. 36, no. 3, pp. 1070–1077, 2013. DOI: 10.1016/j.jnca.2013.01.013

[14] G. Bossert, F. Guihéry, and G. Hiet, "Towards automated protocol reverse engineering using semantic information," *Proceedings of the 9th ACM symposium on Information, computer and communications security*, pp. 51–62, 2014. DOI: 10.1145/2590296.2590346

[15] I. Bermudez, A. Tongaonkar, M. Iliofotou, M. Mellia, and M. M. Munafò, "Towards automatic protocol field inference," *Computer Communications*, vol. 84, pp. 40–51, 2016. DOI: 10.1016/j.comcom.2016.02.015

[16] J. Caballero, H. Yin, Z. Liang, and D. Song, "Polyglot: Automatic extraction of protocol message format using dynamic binary analysis," *CCS '07 Proceedings of the 14th ACM conference on Computer and communications security*, pp. 317–329, 2007. DOI: 10.1145/1315245.1315286

[17] Z. Lin, X. Jiang, D. Xu, and X. Zhang, "Automatic protocol format reverse engineering through context-aware monitored execution," *15th Symposium on Network and Distributed System Security (NDSS)*, 2008.

[18] Z. Wang, X. Jiang, W. Cui, X. Wang, and M. Grace, "ReFormat: Automatic reverse engineering of encrypted messages," *Proceedings of the 14th European Symposium on Research in Computer Security (ESORICS)*, pp. 200–215, 2009. DOI: 10.1007/978-3-642-04444-1_13

[19] C. Leita, K. Mermoud, and M. Dacier, "ScriptGen: an automated script generation tool for Honeyd," *21st Annual Computer Security Applications Conference*, pp. 203–214, 2005. DOI: 10.1109/CSAC.2005.49

[20] P. M. Comparetti, G. Wondracek, C. Kruegel, and E. Kirda, "Prospex: Protocol specification extraction," *30th IEEE Symposium on Security and Privacy*, pp. 110–125, 2009. DOI: 10.1109/SP.2009.14

[21] Y. Wang, Z. Zhang, D. Yao, B. Qu, and L. Guo, "Inferring protocol state machine from network traces: A probabilistic approach," *ACNS 2011: Applied Cryptography and Network Security*, pp. 1–18, 2011. DOI: 10.1007/978-3-642-21554-4_1

[22] C. Y. Cho, D. Babić, P. Poosankam, K. Z. Chen, E. X. Wu, and D. Song, "MACE: Model-inference-assisted concolic exploration for protocol and vulnerability discovery," *SEC'11 Proceedings of the 20th USENIX conference on Security*, 2011.

[23] X. Gao, B. Xiao, D. Tao, and X. Li, "A survey of graph edit distance," *Pattern Analysis and Applications*, vol. 13, no. 1, pp. 113–129, 2010. DOI: 10.1007/s10044-008-0141-y

[24] M. Pawlik and N. Augsten, "Efficient computation of the tree edit distance," *ACM Transactions on Database Systems (TODS)*, vol. 40, no. 1, 2015. DOI: 10.1145/2699485

[25] ——, "Tree edit distance: Robust and memory-efficient," *Information Systems*, vol. 56, pp. 157–173, 2016. DOI: 10.1016/j.is.2015.08.004

[26] Modbus Organization, Inc., "Modbus application protocol specification v1.1b3," http://www.modbus.org/docs/Modbus Application Protocol V1 1b3.pdf, 2012.

[27] A. Banks and R. Gupta, "MQTT Version 3.1.1 (OASIS Standard)," http://docs.oasis-open.org/mqtt/mqtt/v3.1.1/mqttv3.1.1. html, 2014.

**Gergő Ládi** was born in Hungary in 1990. He is a member of the Balatonfüred Student Research Group. He received his Master's degree in Computer Science Engineering in 2018 from Budapest University of Technology and Economics, Hungary, where he is currently pursuing his Ph.D. degree with the Laboratory of Cryptography and System Security. His main areas of research are protocol reverse engineering automation, cloud security, and the security of operating systems.

**Levente Buttyán** received the M.Sc. degree in Computer Science from the Budapest University of Technology and Economics (BME) in 1995, and earned the Ph.D. degree from the Swiss Federal Institute of Technology – Lausanne (EPFL) in 2002. In 2003, he joined the Department of Networked Systems and Services at BME, where he currently holds a position as an Associate Professor and leads the Laboratory of Cryptography and Systems Security (CrySyS Lab). He has done research on the design and analysis of secure protocols and privacy enhancing mechanisms for wireless networked embedded systems (including wireless sensor networks, mesh networks, vehicular communications, and RFID systems). He was also involved in the analysis of some high profile targeted malware, such as Duqu, Flame, MiniDuke, and TeamSpy. His current research interest is in security of cyber-physical systems (including industrial automation and control systems, modern vehicles, cooperative intelligent transport systems, and the Internet of Things in general). Levente Buttyán played instrumental roles in various national and international research projects, published 150+ refereed journal articles and conference/workshop papers, and co-authored multiple books and patents. Besides research, he teaches courses on applied cryptography and IT security at BME and at the Aquincum Institute of Technology (AIT Budapest), and he leads a talent management program in IT security in the CrySyS Lab. He also co-founded multiple spin-off companies, notably Tresorit, Ukatemi Technologies, and Avatao.

**Tamás Holczer** was born in 1981 in Budapest. He received the Ph.D. degree in Computer Science from the Budapest University of Technology and Economics (BME) in 2013. Since 2013 he has been working as an assistant professor in the Laboratory of Cryptography and System Security (CrySyS), Department of Telecommunications, Budapest University of Technology and Economics. Fields of interest: In the past his research interests and his Ph.D. dissertation were focused on the privacy problems of wireless sensor networks and ad hoc networks. Lately he is working on the security aspects of cybe physical systems. The research topics include: security of industrial control networks, honeypot technologies in embedded systems, network monitoring and intrusion detection in industrial networks, and security aspects of intra-vehicular networks.

# Graph construction with condition-based weights for spectral clustering of hierarchical datasets

Dávid Papp[1], Zsolt Knoll[2], and Gábor Szűcs[3]

*Abstract*—**Most of the unsupervised machine learning algorithms focus on clustering the data based on similarity metrics, while ignoring other attributes, or perhaps other type of connections between the data points. In case of hierarchical datasets, groups of points (point-sets) can be defined according to the hierarchy system. Our goal was to develop such spectral clustering approach that preserves the structure of the dataset throughout the clustering procedure. The main contribution of this paper is a set of conditions for weighted graph construction used in spectral clustering. Following the requirements – given by the set of conditions – ensures that the hierarchical formation of the dataset remains unchanged, and therefore the clustering of data points imply the clustering of point-sets as well. The proposed spectral clustering algorithm was tested on three datasets, the results were compared to baseline methods and it can be concluded the algorithm with the proposed conditions always preserves the hierarchy structure.**

*Index Terms*—**spectral clustering, hierarchical dataset, graph construction**

## I. INTRODUCTION

Many clustering methods have been developed, each of which uses a different induction principle [22][29]. Farley and Raftery [8] suggest dividing the clustering methods into two main groups: hierarchical and partitioning methods [25]; and other authors [10] suggest categorizing the methods into additional three main categories: density-based methods [5], model-based clustering [19] and grid-based methods [11]. Partitioning methods are divided into two groups: center-based and graph-theoretic clustering (spectral clustering).

Clusterability for spectral clustering, i.e. the problem of defining what is a "good" clustering, has been studied in some papers [1][2]. HSC [16] algorithm was developed to cluster arbitrarily shaped data more efficiently and accurately by combining spectral and hierarchical clustering techniques. Francky Fouedjio suggested a novel spectral clustering algorithm, which integrates such similarity measure that takes into account the spatial dependency of data, and therefore it is able to discover spatially contiguous and meaningful clusters in multivariate geostatistical data [9]. Furthermore, Li and Huang proposed an effective hierarchical clustering algorithm called SHC [15] that is based on the techniques of spectral clustering method. Although, none of the above studies focus on the case when the input dataset itself is a hierarchical dataset. The spectral clustering method is computationally expensive compared to e.g. center-based clustering, as it needs to store and manipulate similarities (or distances) between all pairs of points instead of only distances to centers [20].

A regular dataset $X = \{x_1, \dots, x_n\}$ consists of $n$ data points and usually there is no pre-defined connection between any two $(x_i, x_j)$ data points. Then clustering $X$ into $k$ clusters can be performed without any restriction on the composition of clusters; this process yields clusters $C_1, \dots, C_k$. On the other hand, a hierarchical dataset designates parent-child relationships between the points (as can be seen in Fig. 1); e.g. $x_i$ and $x_j$ could be the children of $x_l$, so in this case $(x_i, x_j)$ together form a so called *point-set*.

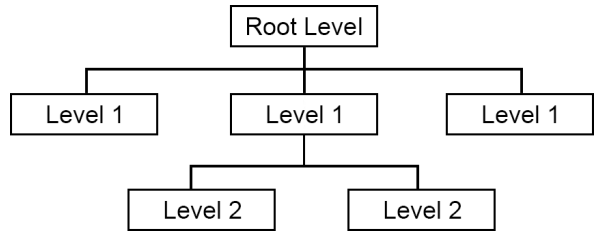



**Figure 1.** Structure of hierarchical dataset

Performing a traditional clustering algorithm also produces the $C_1, \dots, C_k$ clusters, however $x_i$ could be part of $C_g$, while $x_j$ could be assigned to $C_h$, and therefore the $(x_i, x_j)$ point-set would be separated. This means that it is possible that clustering breaks the hierarchical structure of the dataset. In this paper we propose a set of conditions to control the weighted graph creation procedure in the course of spectral clustering [27] algorithm. Using the graph built accordingly will prevent the splitting of point-sets during clustering.

There are several different techniques to build the similarity graph in the spectral clustering, e.g. the ε-neighborhood, k-nearest neighbor and fully connected graphs [27]. The difference between them is how they determine whether two vertices ($x_i$ and $x_j$) are connected by an edge or not. Let us

[1,3]Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary; and Zs. Knoll ([2]) is student in BME Balatonfüred Student Research Group.

[1,3]E-mail: {pappd, szucs}@tmit.bme.hu

DOI: 10.36244/ICJ.2020.2.5

denote the similarity between $x_i$ and $x_j$ by $s_{ij}$; classic spectral clustering method creates a similarity graph $G$, and then proceed as follows:

1. First, a similarity matrix $S$ is derived from $G$, where an $s_{ij}$ element corresponds to the weight of the edge between $x_i$ and $x_j$ in $G$ (in case of not connected points $s_{ij} = 0$).

2. Then diagonal matrix $D$ is calculated by summing the columns of $S$, as can be seen in Eq. 1.

$$D = \{d_{ii}\};\ d_{ii} = \sum_j s_{ij} \tag{1}$$

3. After that the graph Laplacian matrix $L$ is determined from $S$ and $D$ [12], which is a crucial part of spectral clustering, since different $L$ lead to different approach. In this paper the symmetric normalized graph Laplacian is used, which can be computed as expressed in Eq. 2.

$$L_{sym} = D^{-1/2} * S * D^{-1/2} \tag{2}$$

4. Calculate the first $k$ eigenvectors of $L_{sym}$ and then construct a column matrix $U$ from these vectors.

5. Perform K-means clustering on the rows of $U$ to form $C_1, \dots, C_k$.

Majority of authors use graph Laplacian matrix [3][26] in the spectral clustering method, but there is possibility to use other type, so called adjacency matrix [4][14][21]. The eigen decomposition step can be computationally intensive. However, with an appropriate implementation, for example using sparse neighborhood graphs instead of all pairwise similarities, the memory and computational requirements can be solved. Several fast and approximate methods for spectral clustering have been proposed [6][17][28]. The traditional spectral clustering does not make any assumptions about the cluster shapes, but in our research, we dealt with point-sets instead of simple points, so points in a common set are expected to get a common cluster as well.

This concludes the spectral clustering and applying this procedure without any additional modification on a hierarchical dataset would result in a possible structure division. Two novel weight graphs were suggested, the Fully-Connected Weight Graph (FC-WG) and the Nearest Points of Point-sets Weight Graph (NPP-WG) [23]; that can influence the result of spectral clustering algorithms in such way that points belonging to the same point-set will stay together after the clustering is performed. To achieve this behavior the $G$ similarity graph in the original algorithm should be replaced with either FC-WG or NPP-WG. The former is a fully connected graph, where the weight of an edge ($w_{ij}$) between two points ($x_i, x_j$) is calculated according to Eq. 3. Basically the weight is higher in case $x_i$ and $x_j$ are part of the same point-set ($\mathrm{x_i} \leftrightarrow \mathrm{x_j}$), and it is lower if they are not ( $\mathrm{x_i} \nleftrightarrow \mathrm{x_j}$).

$$w_{ij} = \left\{ \begin{matrix} n \\ s_{ij} \end{matrix} \middle| \begin{matrix} \mathrm{x_i} \leftrightarrow \mathrm{x_j} \\ \mathrm{x_i} \nleftrightarrow \mathrm{x_j} \end{matrix} \right\} \tag{3}$$

where $n$ denotes the number of points in the dataset. The NPP-WG is an incomplete graph, because connections between different point-sets are limited, however points that are part of the same point-set still form a fully connected subgraph; as can be seen in Eq. 4.

$$w_{ij} = \left\{ \begin{matrix} n \\ \mathrm{s_{ij}} \\ 0 \end{matrix} \middle| \begin{matrix} \mathrm{x_i} \leftrightarrow \mathrm{x_j} \\ \mathrm{x_i} \nleftrightarrow \mathrm{x_j}\ \&\ \mathrm{s_{ij}} \geq \mathrm{s_{it}}: \forall x_t \left(\mathrm{x_j} \leftrightarrow \mathrm{x_t}, x_j \neq x_t\right) \\ otherwise \end{matrix} \right\} \tag{4}$$

The fundamental idea behind these modifications is to connect any two points inside the same point-set with an increased edge weight that is higher than $s_{ij}$. Although this adjustment does not guarantee that the point-sets remain intact, it only reduces the chance to separate them. The focus of our research was to establish a set of conditions that the weighted graph creation process should satisfy in order to ensure the preservation of point-sets in the hierarchical dataset. In the next section we present the proposed condition system, then Section III contains the result of our experimental evaluation, and in the last section the conclusions of the research are summarized.

## II. SET OF CONDITIONS FOR WEIGHTED GRAPH CONSTRUCTION

With appropriate conditions can be achieved that the points in the same point-set stay together, when using FC-WG and NPP-WG methods. For the formulas the following notations were used:

$n$: $number\ of\ points$
$k$: $number\ of\ clusters$
$C_i$: $i^{th}\ cluster$
$|C_i|$: $number\ of\ datapoints\ in\ the\ i^{th}\ cluster$
$\overline{C_i}$: $complement\ of\ C_i$
$S_i$: $i^{th}\ pointset$
$A$: $similarity\ matrix$
$A_{ij}$: $the\ j^{th} element\ of\ the\ i^{th}\ row\ in\ the\ A\ matrix$
$Z$: $edge\ weigths\ inside\ point\ sets$

The normalized spectral clustering is the relaxation of the normalized cut [26][27]:

$$Ncut(C_1, \dots, C_k) = \sum_{i=1}^{k} \frac{cut(C_i, \overline{C_i})}{vol(C_i)} = \\ = \frac{1}{2} \sum_{i=1}^{k} \frac{\sum_{j \in C_i} \sum_{l \in \overline{C_i}} A_{jl}}{\sum_{j \in C_i} \sum_{l \in \overline{C_i}} A_{jl} + \sum_{j \in C_i} \sum_{l \in C_i} A_{jl}} \tag{5}$$

We investigate two cases of cluster design, and express the formula presented by Eq. 5 in these situations. In the first case we assume that all points in the same point-set is assigned to the same cluster by the clustering algorithm. The second case is when a point (and only one point) was assigned into a different cluster than all other points of the point-set where this particular point belongs to. Note that in the second situation there is only one specific point that is separated from its point-set in the entire dataset.

Let $InC^1$ (inter cluster) be the sum of the edge weights between the clusters, and let $WiC^1$ (within cluster) be the sum of the edge weights inside the clusters; in the first investigated situation, which is denoted by "1" in the superscripts (as can be seen in Eq. 6 and Eq. 7).

$$InC^1(C_i) = \sum_{j \in C_i} \sum_{l \in \overline{C_i}} A_{jl} \tag{6}$$

$$WiC^1(C_i) = \sum_{j \mid S_j \in C_i} \left( \sum_{l \in S_j} \sum_{m \in S_j} Z + \sum_{l \in S_j} \sum_{m \in C_i \setminus S_j} A_{lm} \right) \tag{7}$$

According to Eq. 6 and Eq. 7, $Ncut$ of first case ($Ncut^1$) can be written as:

$$Ncut^1(C_1, \dots, C_k) = \frac{1}{2} \sum_{i=1}^{k} \frac{InC^1(C_i)}{InC^1(C_i) + WiC^1(C_i)} \tag{8}$$

Now let $u$ be the separated point in the second case and $C_k$ its assigned cluster, furthermore denote the cluster which contains all the other points from $u$'s point-set by $C_{\overline{k^u}}$. In this second situation two different inter cluster and two different within cluster aggregates are examined, and the corresponding sub-cases are denoted in the superscripts; e.g. "2,1" refers for the first sub-case of the second situation. Define $InC^{2,1}$ as the sum of edge weights between cluster $C_{\overline{k^u}}$ and any other cluster, while $WiC^{2,1}$ represents the sum of the edge weights within $C_{\overline{k^u}}$; as expressed in Eq. 9 and Eq. 10.

$$InC^{2,1}(C_i, S_t, u) = \sum_{j \in C_{\overline{k^u}}} \sum_{l \in \overline{C_{\overline{k^u}} \cup S_t}} A_{jl} + \sum_{j \in S_t \setminus u} Z + \sum_{j \in \overline{C_{\overline{k^u}} \cup S_t \setminus u}} A_{uj} \tag{9}$$

$$WiC^{2,1}(C_i, S_t, u) = \sum_{j \neq t \mid S_j \in C_i} \left[ \sum_{l \in S_j} \sum_{m \in S_j} Z + \sum_{l \in S_j} \sum_{m \in C_i \setminus S_j} A_{lm} \right] + \sum_{l \in C_i} A_{ul} + Z \tag{10}$$

For the summarized outer and inner edge weights of cluster $C_k$ we introduce $InC^{2,2}$ and $WiC^{2,2}$, respectively; as can be seen in Eq. 11-12.

$$InC^{2,2}(C_i, S_t, u) = \sum_{j \in C_i} \sum_{l \in \overline{C_i \cup S_u}} A_{jl} + \sum_{j \in S_t \setminus u} Z + \sum_{j \in \overline{S_t \setminus u}} A_{uj} \tag{11}$$

$$WiC^{2,2}(C_i, S_t, u) = \sum_{j \neq t \mid S_j \in C_k} \left[ \sum_{l \in S_j} \sum_{m \in S_j} Z + \sum_{l \in S_j} \sum_{m \in C_k \setminus S_j} A_{lm} \right] + \sum_{j \in C_k \setminus u} A_{uj} \tag{12}$$

Based on the above equations $Ncut$ of second case ($Ncut^2$) can be expressed as:

$$\begin{aligned} Ncut^2(C_1, \dots, C_k) = \frac{1}{2} \frac{InC^1(C_i)}{InC^1(C_i) + WiC^1(C_i)} + \\ + \frac{1}{2} \frac{InC^{2,1}(C_{\overline{k^u}}, S_t, u)}{InC^{2,1}(C_{\overline{k^u}}, S_t, u) + WiC^{2,1}(C_{\overline{k^u}}, S_t, u)} + \\ + \frac{1}{2} \frac{InC^{2,2}(C_i, S_t, u)}{InC^{2,2}(C_i, S_t, u) + WiC^{2,2}(C_i, S_t, u)} \end{aligned} \tag{13}$$

We will define the value of $Z$ so that it satisfies the condition that $Ncut^1$ should be lower than $Ncut^2$. To achieve this, we estimated the value of $Ncut^1$ from above, and estimate the value of $Ncut^2$ from below.

In order to estimate $Ncut^1$ from above (see Eq. 16), we substituted $InC^1$ with a larger and replaced the value of $WiC^1$ with a smaller quantity. The substitution in case of $InC^1$ was accomplished by setting the elements of $A$ to 1, and maximizing the number of point-sets, while during the calculation of $WiC^1$ the values of the elements of $A$ were changed to 0, and the number of point-sets was minimized; as can be seen in Eq. 14 and Eq. 15, respectively.

$$InC^1(C_i) \leq n * n * 1 = n^2 \tag{14}$$

$$\begin{aligned} WiC^1(C_i) \geq \sum_{j \mid S_j \in C_i} \left(1^2 * Z + |S_j| \left(|C_i| - |S_j|\right) * 0\right) \geq \\ \geq n * Z \end{aligned} \tag{15}$$

$$\begin{aligned} Ncut^1(C_1, \dots, C_k) \leq \frac{1}{2} \sum_{i=1}^{k} \frac{n^2}{n^2 + n * Z} = \\ = \frac{k * n^2}{n^2 + n * Z} = \frac{k * n}{n + Z} \end{aligned} \tag{16}$$

To estimate the value of $Ncut^2$ from below, the previously defined substitutions were reversed, thus when computing the sum of inner edge weights ($InC^{2,1}$ and $InC^{2,2}$) the matrix $A$ contained only 0 elements, and the number of point-sets was minimized. In accordance with this, the elements of $A$ was set to 1, and the number of point-sets was maximized when $WiC^{2,1}$ and $WiC^{2,2}$ were calculated.

$$InC^1(C_i) \geq \sum_{j \in C_i} \sum_{l \in \overline{C_i}} 0 = 0 \tag{17}$$

$$\begin{aligned} InC^{2,2}(C_i, S_t, u) \geq \\ \geq \sum_{j \in C_{k-1}} \sum_{l \in \overline{C_{k-1} \cup S_t}} 0 + 1 * Z + \sum_{j \in \overline{C_{k-1} \cup S_t \setminus u}} 0 = Z \end{aligned} \tag{18}$$

$$\begin{aligned} InC^{2,2}(C_i, S_t, u) \geq \\ \geq \sum_{j \in C_i} \sum_{l \in \overline{C_i \cup S_u}} 0 + \sum_{j \in S_t \setminus u} Z + \sum_{j \in \overline{S_t \setminus u}} 0 = Z \end{aligned} \tag{19}$$

$$\begin{gathered} WiC^{2,1}(C_i, S_t, u) \leq \\ \leq \sum_{j \neq t | S_j \in C_i} [n^2 * Z + n * n * 1] + (n-1) * 1 + Z \leq \\ \leq n * [n^2 Z + n^2] + n - 1 + Z \leq \\ \leq n^3 Z + n^3 + n \end{gathered} \tag{20}$$

$$\begin{gathered} WiC^{2,2}(C_i, S_t, u) \leq \\ \leq \sum_{j \neq t | S_j \in C_k} [n^2 * Z + n * n * 1] + (n-1) * 1 \leq \\ \leq n * [n^2 Z + n^2] + n - 1 \leq n^3 Z + n^3 + n \end{gathered} \tag{21}$$

$$\begin{gathered} Ncut^2(C_1, \dots, C_k) \geq \\ \geq 0 + \frac{Z}{Z + n^3 * Z + n^3 + n + Z} + \frac{Z}{Z + n^3 * Z + n^3 + n} \geq \\ \geq \frac{2 * Z}{Z + n^3 * Z + n^3 + n + Z} = \frac{2 * Z}{(n^3 + 2) * Z + n^3 + n} \end{gathered} \tag{22}$$

The value of $Ncut^1$ should be lower than $Ncut^2$ in every case. Furthermore, both of them contain a multiplier of ½, and thus it could be eliminated in the equations.

$$Ncut^1(C_1, \dots, C_k) < Ncut^2(C_1, \dots, C_k) \tag{23}$$

$$\frac{k * n}{n + Z} < \frac{2 * Z}{(n^3 + 2) * Z + n^3 + n} \tag{24}$$

$$0 < 2Z^2 + (2n - kn^4 - 2kn)Z - (kn^4 + kn^2) \tag{25}$$

$$Y = \sqrt{k^2 n^8 + 4k^2 n^5 + 4k^2 n^2 + 8kn^4 - 4kn^5 + 4n^2} \tag{26}$$

$$Z < \frac{kn^4 + 2kn - 2n - Y}{4} \tag{27}$$

or

$$Z > \frac{kn^4 + 2kn - 2n + Y}{4} \tag{28}$$

Both (27) and (28) fulfills the conditions in (24) and (25), but the value of (27) is negative in all cases (see Eq. 29, 30 and 31), which means that (27) can not be interpreted as a similarity value.

$$kn^4 + 2kn - 2n - Y < 0 \tag{29}$$

$$\begin{gathered} k^2 n^8 + 4k^2 n^2 + 4n^2 + 4k^2 n^5 - 4kn^5 - 8kn^2 < \\ < k^2 n^8 + 4k^2 n^5 + 4k^2 n^2 + 8kn^4 - 4kn^5 + 4n^2 \end{gathered} \tag{30}$$

$$0 < 8kn^4 + 8kn^2 \tag{31}$$

Based on the above, the similarity value between points in the same point-set should be higher than the $Z_{threshold}$ (see Eq. 32) to avoid the separation of point-sets during spectral clustering. This is only true if the values of the similarity function are between 0 and 1.

$$Z_{threshold} = \frac{kn^4 + 2kn - 2n + Y}{4} \tag{32}$$

Note that $Z_{threshold}$ could be a very large number, even for a reasonably sized dataset, and therefore some sort of normalization of the edge weights is advised to prevent numerical limitations during the matrix manipulations.

## III. EXPERIMENTAL RESULTS

We conducted experiments on three hierarchical datasets to demonstrate the efficiency of the proposed approach. The Free Music Analysis (FMA) audio dataset contains 106,574 tracks from 16,341 artists and 14,854 albums, arranged in a hierarchical taxonomy of 161 genres [7]. The first test dataset composed from the top 12 genres of the hierarchy. To form the second one, the artists were sorted in a decreasing order based on their number of corresponding tracks, and then the top 50 artists were selected. We call the former FMA1 dataset and it contains 9,355 tracks from 1,829 albums, while the latter is called FMA2 dataset, which involves 1,171 albums consist of 10,848 tracks (as can be seen in Table 1). Each track in the FMA collection is represented by a 518-long vector and we used them as input of the spectral clustering algorithm. In this case tracks are equivalent to the points on the lowest level of the hierarchy, while albums are analogous to point-sets.

The third test dataset is a subset of the image collection used in the competition of PlantCLEF 2015 [13]. A total of 91,759 images belongs in this dataset, each of them is a photo of a plant taken from one of the 7 pre-defined types of viewpoint (branch, entire, flower, fruit, leaf, stem and leaf-scan). Images about the same plant are organized into so-called observations, 27,907 plant-observations altogether. The original dataset was filtered in accordance with the provided contextual metadata, thus low quality pictures were discarded. The remaining 26,093 plant images from 9,989 observations form the third test dataset, which is called PCLEF dataset (see Table 1). Furthermore, observations were considered as point-sets and images as points. However, representations were unavailable for PlantCLEF images in the competition, and therefore we extracted visual features from the images to generate so called high-level descriptor vectors. 128 dimensional SIFT (Scale Invariant Feature Transform [18]) features were computed on an image and then they were encoded into 65,536 dimensional Fisher-Vectors [24] based on a codebook of 256 Gaussians.

**Table 1**. Number of points, number of point-sets and number of clusters in FMA1, FMA2 and PCLEF test datasets

| | #points | #point-sets | #clusters |
|---|---|---|---|
| FMA1 | 9,355 | 1,829 | 12 |
| FMA2 | 10,848 | 1,171 | 50 |
| PCLEF | 26,093 | 9,989 | 988 |

Four different graph construction approaches were tested, and their results were evaluated during our experiments. In each

case, other steps of the spectral clustering were identical and only the appropriate graphs were changed, which are the following:

- Fully-Connected Weight Graph using $n$ as edge weights inside the point-sets (FC-WG) [23], where $n$ is the number of the points,
- Nearest Points of Point-sets Weight Graph using $n$ as edge weights inside the point-sets (NPP-WG) [23], where $n$ is the number of the points,
- Fully-Connected Weight Graph using $Z$ as edge weights inside the point-sets (FC-WG(Z)),
- Nearest Points of Point-sets Weight Graph using $Z$ as edge weights inside the point-sets (NPP-WG(Z)).

Table 3 shows the result got on all three test datasets using each of the four different weighted graphs (note that "#ps" stands for "number of point-sets" in the second column). As can be seen, by satisfying the proposed condition, both FC-WG(Z) and NPP-WG(Z) were able to retain all of the point-sets throughout the spectral clustering. On the other hand, FC-WG and NPP-WG methods were unable to preserve the hierarchical structure in each case. Based on these results we conclude that the condition of setting the weights (inside point-sets) to at least the value of $Z_{threshold}$ guarantees that clustering the points on the lowest level of the hierarchy implies the clustering of the point-sets as well, without breaking them apart.

**Table 2**. The result of the number of separated point-sets during the spectral clustering of FMA1, FMA2 and PCLEF datasets

| | | #separated point-sets | | | |
|---|---|---|---|---|---|
| | #ps | FC-WG | NPP-WG | FC-WG(Z) | NPP-WG(Z) |
| FMA1 | 1,829 | 2 | **0** | **0** | **0** |
| FMA2 | 1,171 | 43 | 34 | **0** | **0** |
| PCLEF | 9,989 | 11 | **0** | **0** | **0** |

## IV. DISCUSSION

The known clustering methods can group the points in multidimensional space (where the dimensions of the space are the features of the original items, so a point in this space represent the corresponding item in the original reality), but majority of them is not able to group point-sets. In this paper we focused on point-sets (points that are related to each other) instead of only points, where the point-sets can be grouped into larger groups, so a hierarchical structure describes this grouping of data, resulting a hierarchical dataset. We investigated spectral clustering methods in the clustering literature. Our goal was to develop such spectral clustering approach that preserves the structure of the dataset throughout the clustering procedure. The main contribution of this paper was a set of conditions for weight graph construction used in spectral clustering. Following the requirements – given by the conditions – ensures that the hierarchical formation of the dataset remains unchanged, and therefore the clustering of data points imply the clustering of point-sets as well.

The proposed spectral clustering algorithm with graph construction was tested on three datasets and the results were compared to baseline methods. On the first and second datasets, albums with songs (tracks) were clustered, where tracks are equivalent to the points on the lowest level of the hierarchy, while albums are analogous to point-sets. The third dataset consists of pictures of plants. Here the images of plants represent the points, and the species are the point-sets in the hierarchical dataset. On the obtained clusters, we examined the relationships between the points from the point of view of how they reflect the expected structure, thus it was possible to compare different clustering algorithms with different graph construction approaches.

We demonstrated the clustering in hierarchical datasets with two levels, however our method is able to operate in more levels as well. In general, the point-sets should be constructed based on dendrogram (hierarchical tree) of the multi-level dataset. The user selects the required level (the user can choose any level) in this dendrogram, as can be seen in the Fig 2., and the crossing lines determine the point-sets (5 point-sets in the example) with the corresponding leaves of the tree as points.

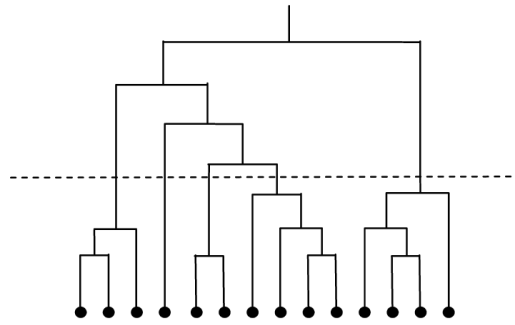


**Figure 2.** Determination the point-sets in hierarchical dataset

We investigated two clustering algorithms: FC-WG (Fully-Connected Weight Graph) and NPP-WG (Nearest Points of Point-sets Weight Graph), where these baseline methods used number of the points ($n$) as edge weights inside the point-sets, during the graph construction. From similarity matrix there are other possibilities to construct a graph, and we elaborated a condition for minimal weight among the points in a common point-set, while other weights come from directly the similarity matrix. So, two graph constructions (a baseline, and the elaborated one with $Z_{threshold}$ value) were investigated in both clustering algorithms, thus four different spectral clustering solutions were in the test: FC-WG, NPP-WG, FC-WG(Z), NPP-WG(Z).

The baseline algorithms using weighted graph approaches, where $n$ values were in the edges, the points in a common point-set did not get into a common cluster; i.e. FC-WG and NPP-WG methods were unable to preserve the hierarchical structure. In the tests, by satisfying the proposed condition, both FC-WG(Z) and NPP-WG(Z) were able to retain all of the point-sets

throughout the spectral clustering. Based on these results we conclude that the condition of setting the weights (inside point-sets) to at least the value of $Z_{threshold}$ guarantees that clustering the points on the lowest level of the hierarchy implies the clustering of the point-sets as well, without breaking them apart.

The developed method is restricted to disjunct point-sets where the point-sets are not overlapping; in the future there is a plan to extend this method to hierarchical datasets with multiple class inheritance as well. The *Z* value influences the clustering result, as can be seen in the comparison with a previous work [23], where *Z* was equal to number of points; further thorough sensitivity analysis of *Z* value is a possible further development in the research.

## ACKNOWLEDGMENT

The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications).

## REFERENCES

[1] Ackerman, M. and Ben-David, S. (2009). Clusterability: A theoretical study. In Dyk, D. A. V. and Welling, M., editors, Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics, AISTATS 2009, Clearwater Beach, Florida, USA, April 16-18, 2009, volume 5 of JMLR Proceedings, pages 1–8. JMLR.org.

[2] Balcan, M. and Braverman, M. (2009). Finding low error clusterings. In COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009

[3] Belkin M. and Niyogi. P. Laplacian eigenmaps for dimensionality reduction and data representation. Neural computation, 15(6):1373-1396, 2003. DOI: 10.1162/089976603321780317

[4] Brand M. and Huang. K. A unifying theorem for spectral embedding and clustering. In Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics, 2003.

[5] Bryant, A., & Cios, K. (2018). RNN-DBSCAN: A density-based clustering algorithm using reverse nearest neighbor density estimates. IEEE Transactions on Knowledge and Data Engineering, 30(6), 1109-1121. DOI: 10.1109/tkde.2017.2787640

[6] Chen, B., Gao, B., Liu, T.-Y., Chen, Y.-F., and Ma, W.-Y. (2006). Fast spectral clustering of data using sequential matrix compression. In Proceedings of the 17th European Conference on Machine Learning, ECML, pages 590–597. DOI: 10.1007/11871842_56

[7] Defferrard, M., Benzi, K., Vandergheynst, P., & Bresson, X. (2016). Fma: A dataset for music analysis. arXiv preprint arXiv:1612.01840.

[8] Farley C. and Raftery A.E., "How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis", Technical Report No. 329. Department of Statistics University of Washington, 1998. DOI: 10.1093/comjnl/41.8.578

[9] Fouedjio, F. (2017). A spectral clustering approach for multivariate geostatistical data. International Journal of Data Science and Analytics, 4(4), 301-312. DOI: 10.1007/s41060-017-0069-7

[10] Han, J. and Kamber, M. Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, 2001.

[11] Hireche, C., Drias, H., & Moulai, H. (2020). Grid based clustering for satisfiability solving. Applied Soft Computing, Vol 88, 106069. DOI: 10.1016/j.asoc.2020.106069

[12] HU, P. (2012). Spectral Clustering Survey.

[13] Joly, A., Müller, H., Goeau, H., Glotin, H, Spampinato, C., Rauber, A., Bonnet, P., Vellinga, W. P., Fisher, B.: LifeCLEF 2015: multimedia life species identification challenges, Proceedings of CLEF 2015 (2015). DOI: 10.1007/978-3-319-24027-5_46

[14] Kannan, R., Vempala, S. and Vetta. A. On clusterings: Good, bad and spectral. Journal of the ACM (JACM), 51(3):497-515, 2004. DOI: 10.1109/sfcs.2000.892125

[15] Li, X., & Huang, J. (2009, November). SHC: a spectral algorithm for hierarchical clustering. In 2009 International Conference on Multimedia Information Networking and Security (Vol. 2, pp. 197-200). IEEE. DOI: 10.1109/mines.2009.107

[16] Liu, L., Chen, X., Luo, D., Lu, Y., Xu, G., & Liu, M. (2013). HSC: A spectral clustering algorithm combined with hierarchical method. Neural Network World, 23(6), 499-521. DOI: 10.14311/nnw.2013.23.031

[17] Liu, T.-Y., Yang, H.-Y., Zheng, X., Qin, T., and Ma, W.-Y. (2007). Fast large-scale spectral clustering by sequential shrinkage optimization. In Proceedings of the 29th European Conference on IR Research, pages 319–330. DOI: 10.1007/978-3-540-71496-5_30

[18] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), 91-110. DOI: 10.1023/b:visi.0000029664.99615.94

[19] McNicholas, P. D. (2016). Model-based clustering. Journal of Classification, 33(3), 331-373.

[20] Meila, M. (2016). Spectral Clustering: a Tutorial for the 2010's. In Handbook of cluster analysis (pp. 1-23). CRC Press.

[21] Ng, A.Y., Jordan, M.I. and Weiss. Y. On spectral clustering: Analysis and an algorithm. Advances in neural information processing systems, 2:849-856, 2002.

[22] Papp, D., & Szűcs, G. (2018). MMKK++ algorithm for clustering heterogeneous images into an unknown number of clusters. ELCVIA Electronic Letters on Computer Vision and Image Analysis, 16(3), 30-45. DOI: 10.5565/rev/elcvia.1054

[23] Papp, Dávid ; Szűcs, Gábor ; Knoll, Zsolt (2019). Machine preparation for human labelling of hierarchical train sets by spectral clustering, Proceedings of the 10th IEEE International Conference on Cognitive Infocommunications (CogInfoCom 2019), pp. 157-162. DOI: 10.1109/coginfocom47531.2019.9089906

[24] Perronnin, F., & Dance, C. (2007, June). Fisher kernels on visual vocabularies for image categorization. In 2007 IEEE conference on computer vision and pattern recognition (pp. 1-8). IEEE. DOI: 10.1109/cvpr.2007.383266

[25] Reddy, C. K., & Vinzamuri, B. (2018). A survey of partitional and hierarchical clustering algorithms. In Data Clustering (pp. 87-110). Chapman and Hall/CRC. DOI: 10.1201/9781315373515-4

[26] Shi J. and Malik. J. Normalized cuts and image segmentation. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 22(8):888-905, 2000.

[27] Von Luxburg, U. (2007). A tutorial on spectral clustering, Statistics and computing, 17(4), 395-416. DOI: 10.1007/s11222-007-9033-z

[28] Wauthier, F., Jojic, N., and Jordan, M. (2012). Active spectral clustering via iterative uncertainty reduction. In 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 1339–1347. DOI: 10.1145/2339530.2339737

[29] Xu, D., & Tian, Y. (2015). A comprehensive survey of clustering algorithms. Annals of Data Science, 2(2), 165-193. DOI: 10.1007/s40745-015-0040-1

**Dávid Papp** was born in 1990 in Hungary and he has received MSc in Computer Science (at specialization of media informatics) from Budapest University of Technology and Economics (BME) in 2016. He started his PhD work in 2016 in the field of Computer Science at the same university. His research topic includes artificial intelligence, machine learning, computer vision as well as development of algorithms on these fields (e.g. query strategies for classification of visual contents with active learning). He was awarded twice with the scholarship of New National Excellence Program of the Ministry of Human Capacities, in 2018 and 2019.

**Zsolt Knoll** was born in Szigetvár, Hungary in 1998. He is a BSc student at Budapest University of Technology and Economics, Faculty of Electrical Engineering and Informatics as computer engineering. He is a member of the Balatonfüred Student Research Group. In focus his research activities are data analytics and machine learning. He took second place in the Students' Scientific Conference at Budapest University of Technology and Economics.

**Gábor Szűcs** has received MSc in electrical engineering and PhD in computer science from Budapest University of Technology and Economics (BME) in 1994 and in 2002, respectively. He is an associate professor at Department of Telecommunications and Media Informatics of BME. His research areas are data science, artificial intelligence, deep learning, content-based image retrieval, multimedia mining. The number of his publications is more than 100. He is the president of the Artificial Intelligence Section of HTE (Scientific Association for Infocommunications), he is the leader of the research group DCLAB (Data Science and Content Technologies). He has earned János Bolyai Research Scholarship of the Hungarian Academy of Science some years ago.

# Performance Analysis of Sparse Matrix Representation in Hierarchical Temporal Memory for Sequence Modeling

Csongor Pilinszki-Nagy[1] and Bálint Gyires-Tóth[2]

***Abstract*—Hierarchical Temporal Memory (HTM) is a special type of artificial neural network (ANN), that differs from the widely used approaches. It is suited to efficiently model sequential data (including time series). The network implements a variable order sequence memory, it is trained by Hebbian learning and all of the network's activations are binary and sparse. The network consists of four separable units. First, the encoder layer translates the numerical input into sparse binary vectors. The Spatial Pooler performs normalization and models the spatial features of the encoded input. The Temporal Memory is responsible for learning the Spatial Pooler's normalized output sequence. Finally, the decoder takes the Temporal Memory's outputs and translates it to the target. The connections in the network are also sparse, which requires prudent design and implementation. In this paper a sparse matrix implementation is elaborated, it is compared to the dense implementation. Furthermore, the HTM's performance is evaluated in terms of accuracy, speed and memory complexity and compared to the deep neural network-based LSTM (Long Short-Term Memory).**

***Index Terms*—neural network, Hierarchical Temporal Memory, time series analysis, artificial intelligence, explainable AI, performance optimization**

## I. INTRODUCTION

Nowadays, data-driven artificial intelligence is the source of better and more flexible solutions for complex tasks compared to expert systems. Deep learning is one of the most focused research area, which utilizes artificial neural networks. The complexity and capability of these networks are increasing rapidly. However, these networks are still 'just' black (or at the best grey) box approximators for nonlinear processes.

Artificial neural networks are loosely inspired by neurons and there are fundamental differences [1], that should be implemented to achieve Artificial General Intelligence (AGI), according to Numenta [2], [3]. [3] They are certain that AGI can only be achieved by mimicking the neocortex and implementing those fundamental differences in a new neural network model.

Artificial neural networks require massive amount of computational performance to train the models through many putational performance to train the models through many epochs. Also, the result of a neural network training is not, or only partly understandable, it remains a black (or at best a grey) box system. There is a need to produce explainable AI solutions, that can be understood. Understanding and modeling the human brain should deliver a better understanding of the decisions of the neural networks.

Sequence learning is a domain of machine learning that aims to learn sequential and temporal data, and time series. Through the years there were several approaches to solve sequence learning. The state of the art deep learning solutions use one-dimensional convolutional neural networks [4], recurrent neural networks with LSTM type cells [5], [6] and dense layers with attention [7]. Despite the improvements over other solutions these algorithms still lack some of the preferable properties, that would make them ideal for sequence learning [1]. The HTM network utilizes a different approach.

Since the HTM network is sparse by nature, it is desirable to implement it in such a way that exploits the sparse structure. Since other neural networks work using optimized matrix implementations, a sparse matrix version is a viable solution to that. This porting should be a two-step process: first a matrix implementation of the HTM network, then a transition to sparse variables inside the network. These ideas are partially present in other experiments, still, this approach remains a unique way of executing HTM training steps. Our goal is to realize and evaluate an end-to-end sparse solution of the HTM network, which utilizes optimized (in terms of memory and speed) sparse matrix operations.

The contributions of this paper are the following:

- Collection of present HTM solutions and their specifics
- Proposed matrix solution for the HTM network
- Proposed sparse matrix solution for the HTM network
- Evaluation of training times for every part of the HTM network
- Evaluation of training times compared to LSTM network
- Evaluation of training and testing accuracy compared to LSTM network

## II. BACKGROUND

There have been a number of works on different sequence learning methods (e.g., Hidden Markov Models [8], Autore-

[1] Balatonfüred Student Research Group

[1,2]Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics Budapest, Hungary

[3]Numenta is a nonprofit research group dedicated to developing the Hierarchical Temporal Memory.

E-mail: csongor.pilinszkinagy@gmail.com; toth.b@tmit.bme.hu

DOI: 10.36244/ICJ.2020.2.6

gressive Integrated Moving Average (ARIMA) [9]), however, in this paper artificial neural network-based solutions are investigated.

### A. Deep learning-based sequence modeling

Artificial neural networks evolved in the last decades and were popularized again in the last years, thanks to the advances in accelerated computing, novel scientific methods and the vast amount of data. The premise of these models is the same: build a network using artificial neurons and weights, that are the nodes in layers and weights connecting them, correspondingly. Make predictions using the weights of the network, and backpropagate the error to optimize weight values based on a loss function. This iterative method can achieve outstanding results [10], [11].

Convolutional neural networks (CNN) utilize the spatial features of the input. This has great use for sequences, since it is able to find temporal relations between timesteps. This type of network works efficiently by using small kernels to execute convolutions on sequence values. The kernels combined with pooling and regularization layers proved to be a powerful way to extract information layer by layer from sequences [12], [4].

Recurrent neural networks (RNN) use previous hidden states and outputs besides the actual input for making predictions. Baseline RNNs are able only to learn shorter sequences. The Long Short-Term Memory (LSTM) cell can store and retrieve the so called inner state and thus, it is able to model longer sequences [13]. Advances in RNNs, including hierarchical learning and attention mechanism, can deliver near state-of-the-art results [14], [15], [16]. An example of advanced solutions using LSTMs is the Hierarchical Attention Network (HAN) [17]. This type of network contains multiple layers of LSTM cells, which model the data on different scopes, and attention layers, which highlight the important parts of the representations.

Attention mechanism-based Transformer models achieved state-of-the-art results in many application scenarios [7]. However, to outperform CNNs and RNNs, a massive amount of data and tremendous computational performance are required.

### B. Hierarchical Temporal Memory

Hierarchical Temporal Memory (HTM) is a unique approach to artificial intelligence that is inspired from the neuroscience of the neocortex [1]. The neocortex is responsible for human intelligent behavior. The structure of the neocortex is homogeneous and has a hierarchical structure where lower parts process the stimuli, and higher parts learn more general features. The neocortex consists of neurons, segments, and synapses. There are vertical connections that are the feedforward and feedback information between layers of cells and there are horizontal connections that are the context inputs. The neurons can connect to other nearby neurons through segments and synapses.

HTM is based on the core assumption that the neocortex stores and recalls sequences. These sequences are patterns of the Sparse Distributed Representation (SDR) input, which are translated into the sequences of cell activations in the network. This is an online training method, which doesn't need multiple epochs of training. Most of the necessary synapse connections are created during the first pass, so it can be viewed as a one-shot learning capability. The HTM network can recognize and predict sequences with such robustness, that it does not suffer from the usual problems hindering the training of conventional neural networks. HTM builds a predictive model of the world, so every time it receives input, it is attempting to predict what is going to happen next. The HTM network can not only predict the future values of sequences but e.g., detect anomalies in sequences.

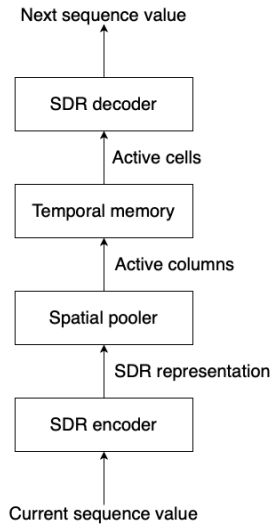



Fig. 1. HTM block diagram

The network consists of four components: SDR Encoder, Spatial Pooler, Temporal Memory, and SDR decoder (see Figure 1).

The four components do the following:

- The SDR Scalar Encoder receives the current input value and represent it an SDR. An SDR representation is a binary bit arrays that retains the semantic similarity between similar input values by overlapping bits.
- The Spatial Pooler activates the columns given the SDR representation of the input. The Spatial Pooler acts as a normalization layer for the SDR input, which makes sure the number of columns and the number of active columns stay fixed. It also acts as a convolutional layer by only connecting to specific parts of the input.
- The Temporal Memory receives input from the Spatial Pooler and does the sequence learning, which is expressed in a set of active cells. Both the active columns and active cells are sparse representations of data just as the SDRs. These active cells not only represent the input data but provide a distinct representation about the context that came before the input.

- The Scalar Decoder takes the state of the Temporal Memory and treating it as an SDR decodes it back to scalar values.

*1) Sparse Distributed Representation:* The capacity of a dense bit array is 2 to the power of the number of bits. It is a large capacity coupled with low noise resistance.

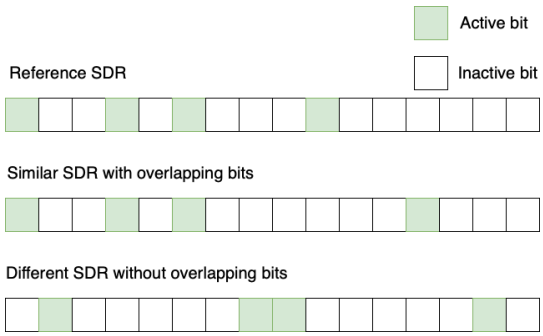



Fig. 2. SDR matching

A spare representation bit array has smaller capacity but is more robust against noise. In this case the network has a 2% sparsity, which means, that only the 2% of columns are activated [1]. A sparse bit array can be stored efficiently by only storing the indices of the ones.

To enable classification and regression there needs to be a way to decide whether or not two SDRs are matching. An illustration for SDR matching can be found in Figure 2. If the overlapping bits in two SDRs are over the threshold, then it is considered as a match. The accidental overlaps in SDRs are rare so the matching of two SDRs can be done with high precision. The rate of a false positive SDR matching is meager.

*2) Encoder and decoder:* The HTM network works exclusively with SDR inputs. There needs to be an encoder for it so that it can be applied to real-world problems. The first and most critical encoder for the HTM system is the scalar encoder. Such an encoder for the HTM is visualized by the Figure 3

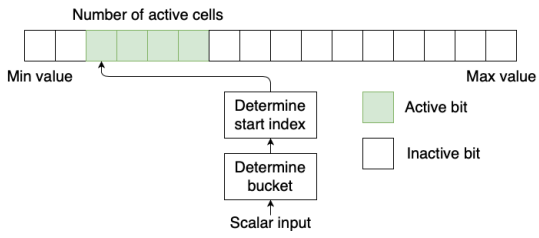



Fig. 3. SDR encoder visualization

The principles of SDR encoding:

- Semantically similar data should result in SDRs with overlapping bits. The higher the overlap, the more the similarity.
- The same input should always produce the same output, so it needs to be deterministic.
- The output should have the same dimensions for all inputs.
- The output should have similar sparsity for all inputs and should handle noise and subsampling.

The prediction is the task of the decoder, which takes an SDR input and outputs scalar values. This time the SDR input is the state of the network's cells in the Temporal Memory. This part of the network is not well documented, the only source is the implementation of the NuPIC package [18]. The SDR decoder visualization is presented in Figure 4.

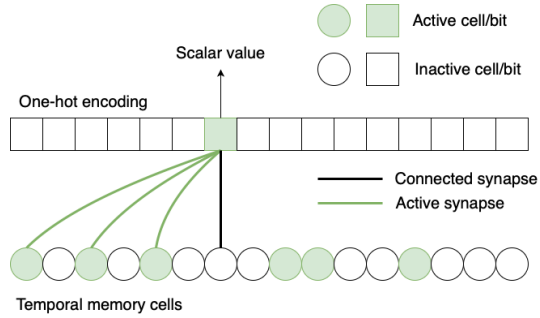



Fig. 4. SDR decoder visualization

*3) Spatial Pooler:* The Spatial Pooler is the first layer of the HTM network. It takes the SDR input from the encoder and outputs a set of active columns. These columns represent the recognition of the input and they compete for activation. There are two tasks for the Spatial Pooler, maintain a fixed sparsity and maintain overlap properties of the output of the encoder. These properties can be looked at like the normalization in other neural networks which helps the training process by constraining the behavior of the neurons.

The Spatial Pooler is shown in Figure 5

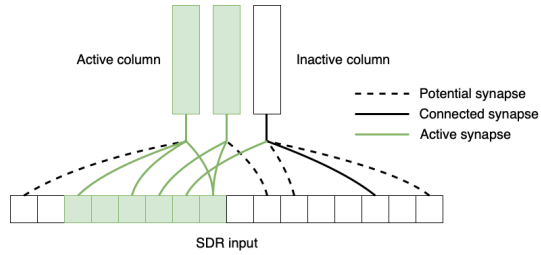



Fig. 5. Spatial Pooler visualization

The Spatial Pooler has connections between the SDR input cells and the Spatial Pooler columns. Every synapse is a potential synapse that can be connected or not depending on its strength. At initialization, there are only some cells connected to one column with a potential synapse. The randomly initialized Spatial Pooler already satisfies the two criteria, but a learning Spatial Pooler can do an even better representation of the input SDRs.

The activation is calculated from the number of active synapses for every column. Only the top 2% is allowed to be activated, the others are inhibited.

*4) Temporal Memory:* The Temporal Memory receives the active columns as input and outputs the active cells which represent the context of the input in those active columns. At any given timestep the active columns tell what the network sees and the active cells tell in what context the network sees it.

A visualization of the Temporal Memory columns cells and connections is provided in Figure 6.

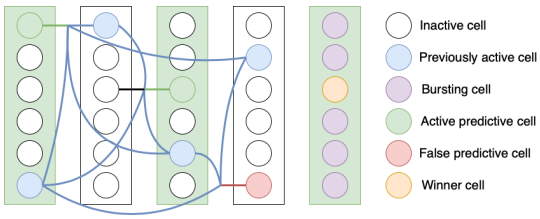



Fig. 6. Temporal Memory connections

The cells in the Temporal Memory are binary, active or inactive. Additionally, the network's cells can be in a predictive state based on their connections, which means activation is anticipated in the next timestep for that cell. The cells inside every column are also competing for activation. A cell is activated if it is in an active column and was in a predictive state in the previous timestep. The other cells can't get activated because of the inhibition.

The connections in the Temporal Memory between cells are created during training, not initialized like in the Spatial Pooler. When there is an unknown pattern none of the cells become predictive in a given column. In this case bursting happens. Bursting expresses the union of all possible context representation in a column, so expresses that the network does not know the context. To later recognize this pattern a winner cell is needed to choose to represent the new pattern the network encountered. The winner cells are chosen based on two factors, matching segments and least used cells.

- If there is a cell in the column that has a matching segment, it was almost activated. Therefore it should be the representation of this new context.
- If there is no cell in the column with a matching segment, the cell with the least segments should be the winner.

The training happens similarly to Spatial Pooler training. The difference is that one cell has many segments connected to it, and the synapses of these segments do not connect to the previous layer's output but other cells in the temporary memory. The training also creates new segments and synapses to ensure that the unknown patterns get recognized the next time the network encounters them.

The synapse reinforcement is made on the segment that led to the prediction of the cell. The synapses of that segment are updated. Also if there were not enough active synapses, the network grows new ones to previous active cells to ensure at least the desired amount of active synapses.

In the case where the cell is bursting the training is different. One cell must be chosen as winner cell. This cell will grow a new segment, which in turn will place the cell in a similar situation into the desired predictive state. The winner cell can be the most active cell, that almost got into predictive state, or the lowest utilized, in other words the cell that has the fewest segments. Only winner cells are involved in the training process. Correctly predicted cells are automatically winner cells as well, so those are always trained. The new segment will connect to some of the winner cells in the previous timestep.

*5) Segments, synapses and training:* In the HTM network segments and synapses connect the cells. Synapses start as potential synapses. This means that a synapse is made to a cell, but not yet strong enough to propagate the activation of the cell. During training, this strength can change and above the threshold the potential synapse becomes connected. A synapse is active if it is connected to an active cell.

The visualization for the segments connection to cells is provided in Figure 7 and the illustration for the synapses connecting to segments is in Figure 8.

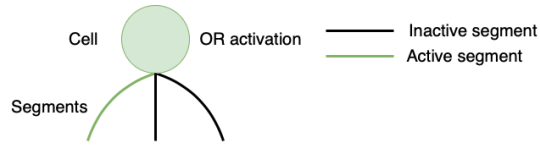



Fig. 7. Segment visualization

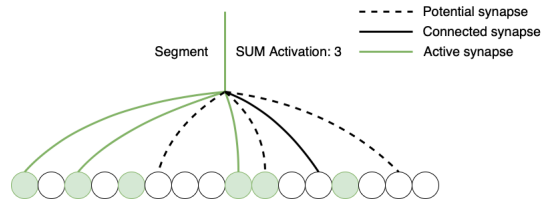



Fig. 8. Synapse visualization

Cells are connected to segments. The segments contain synapses that connect to other cells. A segment's activation is also binary, either active or not. A segment becomes active if enough of its synapses become active, this can be solved as a summation across the segments.

In the Spatial Pooler, one cell has one segment connected to it, so this is just like in a normal neural network. In the Temporal Memory, one cell has multiple segments connected to is. If any segment is activated, the cell becomes active as well. This is like an or operation between the segment activations. One segment can be viewed as a recognizer for a similar subset of SDR representations.

Training of the HTM network is different from other neural networks. In the network, all neurons, segments, and synapses have binary activations. Since this network is binary, the typical loss backpropagation method will not work in this case. The training suited for such a network is Hebbian learning. It is a rather simple unsupervised training method, where the

training occurs between neighboring layers only. The Hebbian learning is illustrated in Figure 9.

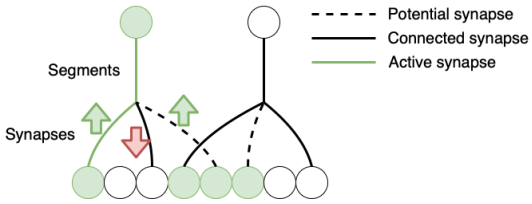



Fig. 9. Visualization of Hebbian learning

- Only those synapses that are connected to an active cell through a segment train.
- If one synapse is connected to an active cell, then it contributed right to the activation of that segment. Therefore its strength should be incremented.
- If one synapse is connected to an inactive cell, then it did not contribute right to the activation of that segment. Therefore its strength should be decreased.

### C. HTM software solutions

There are HTM implementations maintained by Numenta, which give the foundation for other implementations.

First, the NuPIC Core (Numenta Platform for Intelligent Computing) is the C++ codebase of the official HTM projects. It contains all HTM algorithms which can be used by other language bindings. Any further bindings should be implemented in this repository. This codebase implements the Network API, which is the primary interface for creating whole HTM systems. It will implement all algorithms for NuPIC but is currently under transition. The implementation is currently a failing build according to their CircleCI validation [19].

NuPIC is the Python implementation of the HTM algorithm. It is also a Python binding to the NuPIC Core. This is the implementation we choose as baseline. In addition to the other repository's Network API, this also has a High-level API called the Online Prediction Framework (OPF). Through this framework predictions can be made and also it can be also used for anomaly detection. To optimize the network's hyperparameters swarming can be implemented, which generates multiple network versions simultaneously. The C++ codebase can be used instead of the Python implementation if explicitly specified by the user [18].

There is also an official and community-driven Java version of the Numenta NuPIC implementation. This repository provides a similar interface as the Network API from NuPIC and has comparable performance. The copyright was donated to the Numenta group by the author [20].

Comportex is also an official implementation of HTM using Clojure. It is not derived from NuPIC, it is a separate implementation, originally based on the CLA whitepaper [21], then also improved.

Comportex is more a library than a framework because of Clojure. The user controls simulations and can extract useful network information like the set of active cells. These variables can be used to generate predictions or anomaly scores.[4]

There are also unofficial implementations, which are based on the CLA whitepaper or the Numenta HTM implementations.

- Bare Bone Hierarchical Temporal Memory (bbHTM)[3]
- pyHTM[4]
- HTM.core[5]
- HackTM[6]
- HTM CLA[7]
- CortiCL[8]
- Adaptive Sequence Memorizer[9]
- Continuous HTM[10]
- Etaler[11]
- HTM.cuda[12]
- Sanity[13]
- Tiny-HTM[14]

## III. PROPOSED METHOD

The goal of this work is to introduce sparse matrix operations to HTM networks to be able to realize larger models. Current implementations of the HTM network are not using sparse matrix operations, and these are using array-of-objects approach for storing cell connections. The proposed method is evaluated on two types of data: real consumption time-series and synthetic sinusoid data.

The first dataset is provided by Numenta called Hot Gym [22]. It consists of hourly power consumption values measured in kWh. The dataset is more than 4000 measurements long and also comes with timestamps. By plotting the data the daily and weekly cycles are clearly visible.

The second dataset is created by the timesynth Python package producing 5000 data points of a sinusoid signal with Gaussian noise.

A matrix implementation collects the segment and synapse connections in an interpretable data format compared to the array-of-objects approaches. The matrix implementation

[4]Comportex (Clojure), https://github.com/htm-community/comportex, Access date: 14th April 2020

[5]https://github.com/vsraptor/bbhtm, Access date: 14th April 2020

[6]pyHTM, https://github.com/carver, Access date: 14th April 2020

[7]htm.core, https://github.com/htm-community/htm.core, Access date: 14th April 2020

[8]HackTMM, https://github.com/glguida/hacktm, Access date: 14th April 2020

[9]HTM CLA, https://github.com/MichaelFerrier/HTMCLA, Access date: 14th April 2020

[10]ColriCl, https://github.com/Jontte/CortiCL, Access date: 14th April 2020

[11]Adaptive Sequence Memorizer, (ASM),https://github.com/ziabary/Adaptive-Sequence-Memorizer, Access date: 14th April 2020

[12]Continuous HTM GPU (CHTMGPU), https://github.com/222464/ContinuousHTMGPU, Access date: 14th April 2020

[13]Etaler, https://github.com/etaler/Etaler, Access date: 14th April 2020

[14]HTM.cuda, https://github.com/htm-community/htm.cuda, Access date: 14th April 2020

[15]Sanity, https://github.com/htm-community/sanity, Access date: 14th April 2020

[16]Tiny-HTM, https://github.com/marty1885/tiny-htm, Access date: 14th April 2020

achieves the same functionality as the baseline Numenta codebase. The dense matrix implementation has a massive memory consumption, that limits the size of the model. Sparse matrix realization should decrease the required amount of memory.

However, porting to a sparse solution is not straightforward since the support of efficient sparse operations is far less than regular linear algebra.

### A. System design and implementation

In this section the implemented sparse HTM network design is introduced. Throughout the implementation the sparse Python package Scipy.sparse was used, so first that package is described in detail. Next, the four layers of the network are presented, namely the SDR Scalar Encoder, the Spatial Pooler, the Temporal Memory, and the SDR Scalar Decoder. The detailed sparse implementations of these submodules are described, with the matrix implementation in the focus. In this part the sparse matrix and sparse tensor realizations are also discussed.

### B. Matrix implementation

As an initial step, a dense matrix implementation of the NuPIC HTM network was designed and created, which allows treating the HTM network the same as the other widely used and well-optimized networks. While this step was necessary for comparison, it is not suitable for large dataset, since the size of these networks is much bigger compared to the LSTM or CNN networks.

*1) Spatial Pooler:* The network interprets every input into SDR representations which are represented as binary vectors. Multiple inputs can be represented as binary matrices.

The Spatial Pooler columns are connected to the SDR encoder through one segment and its synapses. These connections can be expressed with a matrix, where every row represents an input cell and every column represents a Spatial Pooler column.

The synapse connections have strengths but are used in a binary fashion based on synapse thresholds. Using the binary input vector and the binary connection matrix the column activations are calculated using matrix multiplication. The active column are the ones with the top 2% activation.

*2) Temporal Memory:* In the Temporal Memory cells are connected with other cells. In addition one cell can have multiple segments, so there needs to be a matrix representing every cells connections. For all the cells this results in a tensor, the dimensions are the number of cells along two axis, and the number of maximal segments per cell.

The calculation of cell activation has an extra step, because of the multiple segments. First, the segment activation are calculated for every cell using binary matrix multiplication just as in the Spatial Pooler. These results combined are a matrix, which dimensions are the number of cells times the number of segments. After the activations are calculated, those segments are activated that have above threshold activation values. This results in a binary matrix. Then the cells that are set to be in predictive state are the ones with at least one active segment, with an or operation along the segment axis.

### C. Sparse implementation

Using sparse matrices enables to better scale the network compared to the dense matrix representation. In order to introduce sparse matrix operations to the HTM we used the Scipy.sparse Python package. However, there are missing tensor operators, which were required to be implemented.

There are multiple ways of implementing a sparse matrix representation – different formats can be used for different use-cases. There is the compressed sparse row representation (CSR). The row format is optimal for row-based access in multiplying from the left. The pair of this format is the compressed sparse column format (CSC), which is optimized for column reading, like in right multiplication. From these, the linked list format is beneficial, because it enables the insertion of elements. In the other two cases, insertion is a costly operation.

*1) Sparse matrix:* The network uses the Scipy.sparse package as the main method for matrix operations. This package is extended for further use in the HTM network and also to implement the sparse tensor class. It involves all the common sparse matrix format, which are efficient in memory complexity and have small overhead in computational complexity. This computational complexity decreases as the matrix becomes sparse enough (for matrix dot product around 10% is the threshold).

The realized SparseMatrix class is a wrapper for the Scipy.sparse Python module, extended with operators needed for the Spatial Pooler like reshaping, handling of binary activation matrices, logical operators along axis and random element insertions.

*2) Sparse tensor:* Scipy does not have a sparse tensor implementation. In our case the solution is a dictionary of sparse matrices stacked together. The third dimension is also sparse, it only has a sparse matrix at a given index if it contains at least one nonzero value. The SparseTensor class uses the SparseMatrix class, implementing the same operators.

After all the sparse implementation of the Spatial Pooler and the Temporal Memory differ only in the used classes, since the interfaces are shared across the two. The Spatial Pooler uses sparse vectors as inputs and sparse matrices to store and train the connections. The Temporal Memory receives the input in sparse vectors and stores and trains the connections using sparse tensors.

## IV. EVALUATION AND RESULTS

We carried out experiments on two levels: on operation and network levels. On operation level the dense and sparse realizations were compared in terms of speed and memory capacity, while on the network level the training times and modeling performance of different architectures were compared. In the latter case four different networks were investigated: LSTM, NuPIC HTM, dense HTM, and sparse HTM networks. The baseline LSTM network consists of an LSTM layer with 100

cells and a dense layer also with 100 cells. The training data was generated in autoregressive nature, i.e. with a receptive field of 100 timesteps the network should predict the next instance. The NuPIC HTM network consists of four modules: an SDR Encoder, a Spatial Pooler, a Temporal Memory, and an SDR Decoder(/Classifier). These are configured as the default sizes by Numenta as having 2048 columns, 128 cells per column and 128 maximum segments per cell in the Temporal Memory.

### A. Performance test of the operations

In order to understand the efficiency of the sparse implementation we investigated the scenarios in which the HTM network utilizes sparse matrices. That involves the creation of matrices, element wise product, dot-product, addition, subtraction, and greater or less than operations. The measurements were carried out using randomly generated matrices and tensors at fixed sparsities.

The measurements shown in Table I are carried out on CPU (Intel Core i5, 2 cores, 2GHz)and each represent an average of 1000 runs. Both the dense and sparse matrices are 1000x1000 in size with sparsity of 0.1%.

TABLE I
DENSE AND SPARSE ARITHMETIC MATRIX OPERATION EXECUTION TIMES ON CPU (1000 SAMPLE AVERAGE)

| Operation | Dense time | Sparse time |
|---|---|---|
| Addition | 0.001080s | 0.000418s |
| Subtraction | 0.001104s | 0.000413s |
| Dot-product | 0.0545s | 0.0012s |
| element wise product | 0.001074s | 0.000463s |
| Greater than | 0.000672s | 0.000252s |
| Less than | 0.000649s | 0.049147s |

It is clear that the sparse version has an advantage for almost all operators, however, the ”less than” operator lags behind compared to the dense counterpart. This is because the result has all the values set to true, which is not ideal for a sparse representation. (true being a nonzero value) Still the execution stores this as a sparse matrix which has a measurable overhead.

Next, to understand the efficiency of the sparse tensors we measured the actual scenarios in which the HTM network uses sparse matrices. That is the creation of tensors, element wise product, dot-product, addition, subtraction, and greater or less than operations.

The measurements are shown in Table II with the same settings as before. Both the dense and sparse tensors' shape is 10x1000x1000 with sparsity of 1%.

These results show, that the sparse tensor is slower in cases of addition and subtraction and element wise product. However, this solution excels in dot product, which is the operator needed to calculate activations for a given input. The speedup here compared to the dense implementation is more than a 1000 times. In the case of the less than operator the implementation is also slower due to the same reasons as with the sparse matrices.

TABLE II
DENSE AND SPARSE ARITHMETIC TENSOR OPERATION EXECUTION TIMES ON CPU (1000 SAMPLE AVERAGE)

| Operation | Dense time | Sparse time |
|---|---|---|
| Addition | 0.0040s | 0.0079s |
| Subtraction | 0.0037s | 0.0069s |
| Dot-product | 33.29s | 0.0262s |
| element wise product | 0.0034s | 0.0074s |
| Greater than | 0.0022s | 0.0038s |
| Less than | 0.0015s | 0.3393s |

### B. Performance test of HTM modules

In Table III each different module of the two networks were measured on CPU. Each measurement represent an average over 100 samples.

TABLE III
NUPIC AND SPARSE HTM MODULE EXECUTION TIMES ON CPU (100 SAMPLE AVERAGE)

| Part of the network | NuPIC | Sparse HTM |
|---|---|---|
| SDR Encoder | 0.000303s | 0.000607s |
| Spatial Pooler | 0.0179s | 0.0139s |
| Temporal Memory | 0,0136s | 1.03s |
| SDR Decoder | 0.000303s | 0.24s |

These results show that the proposed sparse implementation has an advantage in the Spatial Pooler, where the inference is more straightforward, so the sparse multiplication speedup can be utilized. However, the Temporal Memory solution still lags behind in execution time.

For the memory complexity the following network sizes are applicable. These numbers are recommendations of Numenta, and are based on their research. The number of columns is a minimum requirement because of the sparse activation. The other parameters have a specific capacity that can be further fitted to a specific need. In general these values should work without the network becoming too big in size and capacity. The SDR size should be 100 with 9 active elements, the network size is 2048 columns, 32 cells per column and 128 segments per cell. The activation should be 40 columns in each timestep. The values in Table IV are measured in the number of integer values stored.

TABLE IV
MEMORY COMPLEXITY OF THE DIFFERENT NETWORK PARTS

| Part of the network | Dense HTM | Sparse HTM |
|---|---|---|
| SDR Encoder output | 100 | 27 |
| Spatial Pooler connections | 204800 | 12000 |
| Spatial Pooler output | 2048 | 120 |
| Temporal Memory connections | $5.49 * 10^{11}$ | $3.35 * 10^{8}$ (max) |
| Temporal Memory output | 65536 | 40-1280 |
| SDR Decoder connections | 6553600 | 65536 (max) |

It is clear that, the sparse solution makes it possible to store the network in a matrix format, since the Temporal Memory

part can easily exceed current hardware limitations. In the case of NuPIC HTM the memory complexity estimation is harder, since that uses array-of-objects structure.

TABLE V
TRAINING TIMES

| Configuration | Epochs | Timesynth | Hot Gym |
|---|---|---|---|
| LSTM | 100 epochs | 243s | 318s |
| NuPIC HTM | 1 epoch | 130s | 138s |
| NUPIC HTM | 5 epochs | 1706s | 1028s |

Last, the LSTM and HTM network predictions for different datasets were investigated. The training times are shown in Table V for the LSTM and NuPIC HTM solutions.

First, in Figure 10 the predictions for both LSTM and HTM can be seen on the test part of Hot Gym dataset for the first 100 elements. The y-axis represents the power consumption of the building. The figure presents the difference between the LSTM and HTM predictions, where LSTM is less noisy and it seems that it gives near naive predictions in some cases. The HTM tends to follow better rapid changes.

In the case of train and test losses the HTM network is not on par with the performance of the LSTM network. In Table VI the performances are evaluated using the Hot Gym test dataset. It shows that the LSTM maintains lower train and test loss values than the HTM network. However, based on the possible interval range and looking at the predictions it is not clear that the HTM network is worse at predicting this dataset (see Figure 10). On the other dataset the achieved results are summarised in Table VII. In this case also the LSTM has a lower training and testing mean squared error. Looking at the predictions the network completely filters out the high frequency part of the data and only retains the base sinusoid signal in the predictions (see Figure 11).

TABLE VI
MINIMUM MSE VALUES FOR HOT GYM DATASET

| Configuration | Epochs | Train MSE | Test MSE |
|---|---|---|---|
| LSTM | 100 | 0.1073 | 0.1658 |
| NuPIC HTM | 5 | 0.3762 | 0.4797 |

TABLE VII
MINIMUM MSE VALUES FOR TIMESYNTH DATASET

| Configuration | Epochs | Train MSE | Test MSE |
|---|---|---|---|
| LSTM | 100 | 0.1603 | 0.1474 |
| NuPIC HTM | 5 | 0.4940 | 0.5069 |

In Figure 11 the predictions for both networks can be seen on the synthetic (Timesynth) test dataset for the first 200 timesteps. In this case it is even more pronounced that the LSTM network smooths the rapid changes in its predictions. There is also a lag its predictions, that is seen as a shift in the direction of the x-axis.

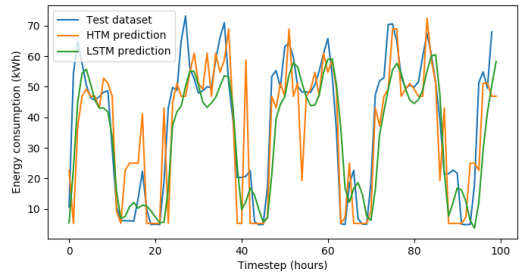



Fig. 10. Predictions for Hot Gym dataset

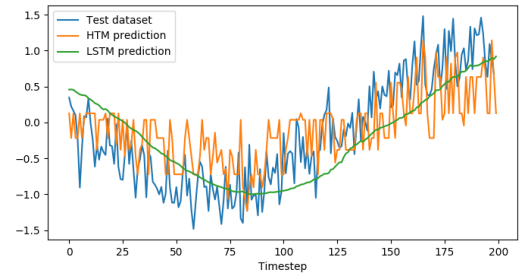



Fig. 11. Predictions for synthetic (Timesynth) data

In Figure 12 the outputs of HTM network are presented after the first and last epoch of training on the Hot Gym test dataset. It shows that the network is able to follow the main cycles of the data from the first epoch. On the other hand in Figure 13 this kind of progress is not that clear.

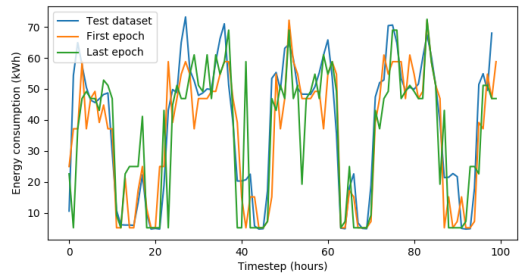



Fig. 12. Difference between first and last epoch for HTM on Hot Gym dataset

## V. CONCLUSIONS

In this paper we investigated the sequence learning possibilities of HTM network. The advantages and disadvantages of different implementations surrounding the HTM network were described and different HTM versions and an LSTM were evaluated on a synthetic sequential dataset and on a real time-series. A methodology of turning the implementation of the HTM network to sparse matrix operations was proposed for lower memory usage. We showed that the proposed methodology is feasible, it uses at least an order of magnitude less

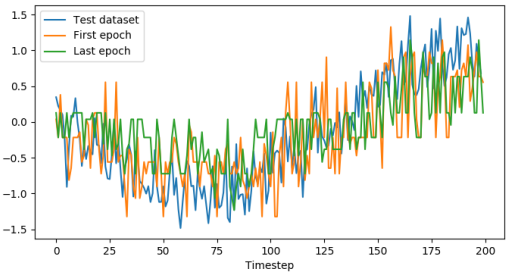



Fig. 13. Difference between first and last epoch for HTM on synthetic (Timesynth) dataset

memory than the dense implementation in the case where the sparsity of the network is at 2%. Furthermore, the proposed method's performance remains comparable to the other HTM implementation.

## ACKNOWLEDGMENT

The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications), by the BME-Artificial Intelligence FIKP grant of Ministry of Human Resources (BME FIKP-MI/SC), by Doctoral Research Scholarship of Ministry of Human Resources (ÚNKP-19-4-BME-189) in the scope of New National Excellence Program and by János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

## REFERENCES

[1] J. Hawkins, S. Ahmad, S. Purdy, and A. Lavin, "Biological and machine intelligence (bami)," 2016, initial online release 0.4. [Online].Available: http://numenta.com/biological-and-machine-intelligence/

[2] Numenta, "Numenta webpage," https://numenta.com, 2019.

[3] ——, "Htm school," https://www.youtube.com/playlist?list=PL3yXMgtrZmDqhsFQzwUC9V8MeeVOQ7eZ9, 2018.

[4] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[5] M. Sundermeyer, R. Schlüter, and H. Ney, "Lstm neural networks for language modeling," in *Thirteenth annual conference of the international speech communication association*, 2012.

[6] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, nov 1997. [Online]. Available: DOI: 10.1162%2Fneco.1997.9.8.1735

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[8] S. R. Eddy, "Hidden markov models," *Current opinion in structural biology*, vol. 6, no. 3, pp. 361–365, 1996.

[9] J. Contreras, R. Espinola, F. Nogales, and A. Conejo, "ARIMA models to predict next-day electricity prices," *IEEE Transactions on Power Systems*, vol. 18, no. 3, pp. 1014–1020, aug 2003. [Online]. Available: DOI: 10.1109%2Ftpwrs.2002.804943

[10] P. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990. [Online]. Available: DOI: 10.1109%2F5.58337

[11] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in *Neural networks: Tricks of the trade*. *Springer*, 2012, pp. 9–48.

[12] Y. LeCun, Y. Bengio et al., "Convolutional networks for images, speech, and time series," *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[13] S. Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 06, no. 02, pp. 107–116, apr 1998. [Online]. Available: DOI: 10.1142%2Fs0218488598000094

[14] J. Koutnik, K. Greff, F. Gomez, and J. Schmidhuber, "A clockwork rnn," *arXiv preprint arXiv:1402.3511*, 2014.

[15] J. Chung, S. Ahn, and Y. Bengio, "Hierarchical multiscale recurrent neural networks," *arXiv preprint arXiv:1609.01704*, 2016.

[16] S. Merity, "Single headed attention rnn: Stop thinking with your head," arXiv preprint arXiv:1911.11423, 2019.

[17] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. *Association for Computational Linguistics*, 2016. [Online]. Available: DOI: 10.18653%2Fv1%2Fn16-1174

[18] Numenta, "Nupic (python)," http://github.com/numenta/nupic, 2019.

[19] ——, "Nupic core (c++)," http://github.com/numenta/nupic.core, 2019.

[20] ——, "Htm.java," http://github.com/numenta/htm.java, 2019.

[21] ——, "Htm cortical learning algorithms," https://numenta.org/resources/HTM_CorticalLearningAlgorithms.pdf, 2011.

[22] ——, Hot Gym dataset, 2024 (accessed July 1, 2020). [Online]. Available: https://github.com/numenta/nupic/blob/master/examples/opf/clients//hotgym/prediction/one_gym/rec-center-hourly.csv

IEEEbiograph

**Csongor Pilinszki-Nagy** conducts research on artificial general intelligence methods since 2016. His work consists of solutions using image recognition by convolutional neural networks, building general solutions in game environments using reinforcement learning and researching unconventional methods like Hierarchical Temporal Memory network. He obtained his MSc degree in Computer Science from the Budapest University of Technology and Economics in January 2020. He is a member of the Balatonfüred Student Research Group.

**Bálint Gyires-Tóth** conducts research on fundamental and applied machine learning since 2007. With his leadership, the first Hungarian hidden Markovmodel based Text-To-Speech (TTS) system was introduced in 2008. He obtained his PhD degree from the Budapest University of Technology and Economics with summa cum laude in January 2014.
Since then, his primary research field is deep learning. His main research interests are sequential data modeling with deep learning and deep reinforcement learning. He also participates in applied deep learning projects, like time series classification and forecast, image and audio classification and natural language processing. He was involved in various successful research and industrial projects, including finance, fraud detection and Industry 4.0. In 2017 he was certified as NVidia Deep Learning Institute (DLI) Instructor and University Ambassador.

# De-anonymizing Facial Recognition Embeddings

István Fábián[1] and Gábor György Gulyás[2]

***Abstract*—Advances of machine learning and hardware getting cheaper resulted in smart cameras equipped with facial recognition becoming unprecedentedly widespread worldwide. Undeniably, this has a great potential for a wide spectrum of uses, it also bears novel risks. In our work, we consider a specific related risk, one related to face embeddings, which are machine learning created metric values describing the face of a person. While embeddings seems arbitrary numbers to the naked eye and are hard to interpret for humans, we argue that some basic demographic attributes can be estimated from them and these values can be then used to look up the original person on social networking sites. We propose an approach for creating synthetic, life-like datasets consisting of embeddings and demographic data of several people. We show over these ground truth datasets that the aforementioned re-identifications attacks do not require expert skills in machine learning in order to be executed. In our experiments, we find that even with simple machine learning models the proportion of successfully re-identified people vary between 6.04% and 28.90%, depending on the population size of the simulation.**

***Index Terms*—facial recognition, de-anonymization, machine learning**

## I. INTRODUCTION

We live in times when efficient uses of artificial intelligence and cheap smart technology are exploding. By the spread of smart cameras, applications on facial recognition had become almost ubiquitous in some cities around the world. In some cases we can find the driver reason for this in the security concerns of the public, but face recognition (or FR in short) can be applied to a much broader set of use-cases. Beside identification or authentication of individuals in crowds, it could benefit the society also in criminal detection, searching for lost people, customer behavior analysis, etc. [1].

However, FR technology could be abused and therefore it has the potential to pose risks to individuals, to the society and even to the governmental and business sectors, as well [2]. This puts related ethical issues into the focus. The French data protection authority, the CNIL (French National Commission on Informatics and Liberty) published a recent paper detailing the technical, legal and ethical challenges regarding these applications [3]. The biggest concern probably is how FR is being a part of emerging surveillance technologies [4]. Consequently, several governments made recent attempts in order to regulate the uses of FR technology.

Despite official guidelines for camera surveillance [5], some believe that automated FR breaches GDPR because it fails to meet the requirement for consent by design [6]. The European Commission even considered imposing a temporary ban on using FR in public spaces, which was later discarded [7]. In their white paper released on the 19th February [8], the European Commission rather envisions an approach where companies evaluate their own data processing practices from a risk-based point of view. This is backed up by a recent proposal to conduct an impact assessment analysis when dealing with FR applications [2].

This debate on the ban is also present in the US. While Washington DC just passed facial recognition rules that allow the use of the technology with some restrictions (e.g. government agencies can only use FR software if it's got an application programming interface, and vendors must reveal any reports of bias) [9], San Francisco was the first city to ban FR entirely in public spaces [10]. The unresolved nature of these issues is further confirmed by the Fundamental Rights Agency, who released a paper about the fundamental rights considerations regarding FR [11].

Certain related risks can be associated with the processing and storing of facial imprints. State-of-the-art face imprints are coming from the domain of Deep Metric Learning (DML), in which deep learning techniques are trained to produce descriptive vectors of faces while also considering their similarity [12]. These vectors, or face embeddings, have high similarity when taken from the same person, but have a low similarity score when taken from different people. While these seem as a list of arbitrary numbers to the naked eye, they may contain personal information about the person whose photo was taken. In their recent work, *Mai et al.* showed that the photo itself can be reconstructible from the embedding [13]. In [14] authors argue that it should be an accepted fact that with good accuracy the original sample can be reconstructed from unprotected embeddings. This means that sensitive data could be derived from unprotected templates and other attacks can also be launched based on the reconstruction results. Based on this, it can also be possible to reverse engineer data from face embeddings in order to find out the original identity of the embedding.

In this paper we examine an attack that aims to find out the original identity of face imprints. As the original faces can be partially rebuilt from embeddings, we look at the scenario where the attacker tries to reconstruct demographic data from the embeddings. First, we measure the level of accuracy achievable in predicting age, sex and race from facial embeddings, then we create a synthetic dataset and run the attack from one end to the other. Our results show that predicting these characteristics is indeed possible with alarming accuracy and re-identification attacks can be executed successfully.

The paper is structured as follows. In Section II we discuss how facial recognition works, the privacy risks of processing face embeddings and how re-identification attacks work. Next,

[1] Balatonfüred Student Research Group
[1,2] Department of Automation and Applied Informatics, Budapest University of Technology and Economics, Hungary.
(e-mail: fabian@aut.bme.hu; gabor.gulyas@aut.bme.hu)

DOI: 10.36244/ICJ.2020.2.7

in Section III, we introduce our attacker model. In Section IV we describe how we used different technologies in our research, and following in Sections V-VI we elaborate our results. Finally, Section VII summarizes our work.

## II. RELATED WORK

### A. Facial Recognition

The main motivation behind facial recognition is to make it possible to identify people, e.g. a person from a digital photo or video frame based on the face's unique characteristics. Despite the fact that it has only become widespread in recent years, the technology has been around for decades, although it wasn't as extensively used as today, because it had many open problems that hindered its performance and accuracy, like the lack of enough computational power and training data, which resulted in poor scalability.

However, the first milestone towards automated FR came in 1988 when Sirovich and Kirby came up with the Eigenface approach [15], which applies linear algebra (including principal component analysis) to recognize faces. Basically, it works by creating an average face and multiple so called Eigenfaces based on all faces available in a dataset, and then representing each new face as a vector made up of the coefficients of the linear combination of the average face and the Eigenfaces. Then the similarity between two faces depends on the distance metric between each face's vector, with a small distance corresponding to higher similarity. In 1991, Turk and Pentland further improved the Eigenface approach to also detect faces in images [16]. Since then, it was in the 2010s when FR technology significantly improved due to the usage of machine learning and deep neural networks. This was made possible by the large amount of training data and computing power available.

In our analysis we wanted to work with state-of-the-art facial recognition techniques that are publicly available in Python libraries and that could be run efficiently on a typical smart camera. One of the leading solutions is found in the dlib library [17], which uses the ResNet-34 structure deep neural network from [18], trained on the Labeled Faces in the Wild dataset (LFW) [19]. Another prominent method is implemented in the OpenCV library. This deep convolutional network uses the FaceNet structure [20] that directly maps face images into the Euclidean space using a triplet-based loss function based on large margin nearest neighbor classification (LMNN) [21]. This library achieves a 99.63% accuracy score on the LFW dataset [19].

Both of these techniques produce a 128 long vector of float values. When comparing the two methods, we found that the technique offered by dlib provides a better trade-off regarding less false positives, with a slightly higher rate of false negatives. Therefore we decided to work with it throughout our experiments.

### B. Risks Related to Embeddings

Face embeddings should be considered biometric data by definition provided by the General Data Protection Regulation (Art 4. §14 in [22]): an embedding consists of data points that were extracted from the photo of a person that allow or enable the identification of the data subject. Due to their nature, biometric attributes capture features of the human body that one cannot be changed. Therefore, significant societal and privacy risks arise, which urges the need to analyze the impacts of this technology [2]. As we discussed previously, modern FR works by extracting templates from photos that need to be stored in a database or compared previously stored ones. If we consider the number of people represented in the images X, and the number of people who are part of a database Y, then FR can be used for authentication (X:1 Y:1), identification (X:1 Y:n) or tracking (X:1 Y: no need for a database). Depending on these various use cases, the risks can be more or less severe, e.g., a big central database means higher risks against malicious actors than a smaller database.

Further reasons for concern are that FR is not a perfect technology, risk appear that had been seen previously in automated decision making systems [23]. For example, FR can be discriminatory due to biases built into the technology, or one may find it difficult to explain in details how DML-based facial recognition works or why it had proposed a specific embedding in a certain situation.

Authors in [24] mention two potential threats regarding an attacker's abilities. One of the hazards is to masquerade the template owner, which means using the biometric template for reconstructing a 2D or 3D model of the template owner's face and using that model to trick a FR system. The other is the possibility of the attacker to do cross matching between multiple databases storing biometric templates, because biometrics are mostly immutable and the same or very similar templates could be stored in multiple databases for different applications. These risks motivate the use of biometric template protection (BTP) schemes that transform biometric templates to make their usage and storage safe, while also keeping their utility.

## III. RISK AND ATTACKER MODEL

In our work, we consider re-identification attacks against a database of face embeddings. Since face embeddings are based on the face's unique characteristics and enable reconstructing faces, they may contain hints for demographic information as well. This can contribute to identification attacks.

Re-identification attacks are when an attacker combines multiple data sources to uncover the identities in the anonymous dataset. A common example is a health care provider who publishes data for research purposes after removing any PII (personally identifiable information) such as names, addresses, social security numbers, etc. However, as [25] showed, it can still be possible to re-identify people in that database by linking it with an additional database (e.g. publicly available voter database). Demographic data can be especially vulnerable against re-identification attacks, as [25] showed that the zip code, sex and date of birth provides a unique identifier for 87% of the US population based on census data.

These examples showed that tabular datasets are vulnerable for re-identification. It has been shown that large datasets, where the number of attributes is rather proportional to the

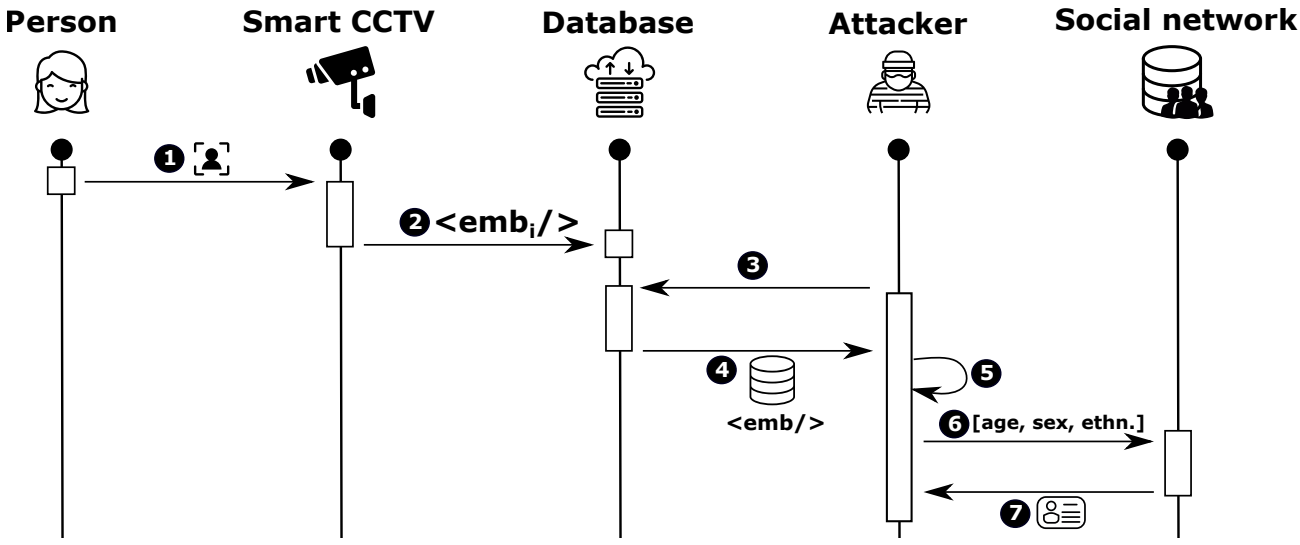



Fig. 1. The considered attack when a malicious third party reconstructs demographic data from embeddings and re-identifies data subjects by linking with another public database.

number of rows, can also be re-identified. Various examples include movie ratings [26], social networks [27], and credit card usage patterns [28]. As explained later, here we consider rebuilding attributes from embeddings that we consider later for re-identification.

In our case, let us consider the following FR system setup that may be deployed at a company, and the corresponding attacker model (see Figure 1). Smart cameras observe the company's various areas and extract the face embedding of employees appearing in the video footage (Step 1). These embeddings are then transferred and stored in a central database for later use either for tracking, automation, identification or other purposes (Step 2). The attacker then accesses these embeddings (Steps 3-4, e.g. en employee by stealing or an external person via hacking) and infers the data subjects' demographic information (age, sex and race) from them using a computer algorithm created for this task (Step 5). With this new information the attacker may now be able to do a successful re-identification attack by comparing the original data with another public data source, for example by looking up people on a social networking site (Steps 6-7).

The success of such an attack largely depends on Step 4 and Step 5 from Figure 1: how many embeddings the attacker can get, and how accurately they can predict demographic information from those embeddings. Thus, it is necessary to assess the potential attacker strength first. In our work, we assume a strong attacker who has access to all the embeddings stored in the database, and our main goal is to discover the level of prediction accuracy achievable regarding demographic data.

## IV. METHODOLOGY

In order to estimate the potential success of attackers, on a real life dataset we considered the equivalence class distribution of demographic details. An equivalence class is a subset of elements that are equivalent to each other based on the demographic characteristics that we are trying to predict. In a database, the more people that are either unique or fall in small equivalence classes (e.g. at most 5 members), the higher

### A. Technical Details

We carried out our analysis in the Python programming language, using open source libraries created for working on data science and machine learning (ML) applications (NumPy [29], pandas [30], Scikit-learn [31]). The face recognition library we used was face_recognition [32], which is a wrapper built around dlib [17] and uses dlib's state-of-the-art FR technology based on deep learning to detect faces in images and/or video frames and extract the face embeddings from them. While embeddings are hard for a human to interpret, a computer can compare two embeddings and calculate the mathematical distance between them, such as Euclidean or Manhattan distance, with the Euclidean distance being the most popular "best practice" choice for face recognition applications. These metrics can be used to determine whether the two embeddings belong to the same person or not. The lower the distance between two embeddings, the more likely it is that they belong to the same person. Usually, there is a distance threshold below which we consider embeddings to belong to the same person.

We used Random Forest Classifiers from the Scikit-learn library to build three ML models for predicting the age, sex and race from the embeddings. We chose a Random Forest Classifier as it is an easy to use ML model that doesn't require hyper parameter tuning and can be used easily even by non ML experts. It is an ensemble-tree based learning algorithm used to predict the class of test objects. Instead of training a single decision tree on the entire training data, the random forest works by training multiple decision trees on randomly sampled subsets of the training set (while also having the attributes randomly distributed), and then aggregating the votes of the

decision trees to conclude the final predicted class by majority voting.

For the data to train and test on, we used UTKFace [33], a public database containing over 23,000 photos from both sexes aged between 1 to over 100, from white, black, asian, indian and other races, where one image per person is included. Due to the fact that the various age, sex and race classes were not balanced, we sampled this data source to gain a more balanced dataset for training and testing (see the following subsection).

### *B. Our Methodology*

Since the biggest majority of the people in UTKFace database are under the age of 80 and are either white, black, asian or indian, we only considered people fitting these constraints. There was a very low number of examples in dropped classes which would have led to poor training and prediction results. However, not all of the remaining classes were balanced. For example there were 2043 photos of white males aged between 20 and 40 years, while only 677 Asian males in the same age range.

So to achieve a relatively balanced training and testing data set, we had to apply data down sampling until we were left with 12192 photos, 1524 photos for each of the 8 race-sex pairs. Yet, the age distribution still was not completely balanced, as there were 2893 people (23.73%) aged between 1 and 20 years, 5515 (45.23%) aged between 21 and 40 years, 2452 (20.11%) aged between 41 and 60 years, while only 1332 people (10.93%) were aged between 61 and 80 years. While we accept this as it is rather life-like, this could hinder model performance. Furthermore, achieving a completely balanced dataset would have resulted in too few examples to train and test with.

The following step is to run the face_recognition library's face_encodings function on all the 12192 images, and storing the face embedding found for each. Since the image file names contain the necessary information about a person's demographics (as all the image file names follow the [age]_[gender]_[race]_[date&time].jpg pattern), the file names were used to create the training labels for each image. Equipped with this labeled data set, it is now possible to use Scikit-learn's RandomForestClassifier class to train a Random Forest Classifier for predicting the age, sex and race from face embeddings. In all models, we found that using a Random Forest of 100 trees can achieve the job (i.e. setting the *n_estimators* parameter to 100). Also, using Scikit-learn train_test_split function to split the data set into 80% training and 20% testing data made it possible to validate our models.

The simplest Random Forest Classifier to train was the one predicting the sex of people based on their face embeddings as this required only binary classification, while predicting the age and race required multi-class classification. Regarding age prediction, expecting the prediction of precise age values resulted in poor performance. First this may sound surprising, but it is impossible even for humans to predict a person's age with such precision. Thus some intervals needed to be defined for age prediction. Choosing narrow age ranges (1-10 years) also resulted in poor prediction accuracy. On the other hand, choosing a too wide age range (25 years and over) would have resulted in very poor utility regarding inference. As a viable trade-off, we divided people into 4 age groups: 1-20, 21-40, 41-60 and 61-80 years.

The results of our experiment are detailed in the following section.

## V. MEASUREMENTS

As seen in Table I, which represents the sex prediction model's confusion matrix on the test data, the model achieved an accuracy score of 91.8%, and an F1 score of 91.8%. Looking at the confusion matrix it can be concluded that even such a simple model can correctly recognize with closely the same accuracy both males and females. Figure 2 shows the receiver operating characteristic (ROC) curve which achieved an area under curve (AUC) value of 97.6%.

Table II shows the confusion matrix of the age prediction model's performance on the test data. It can be seen that the age prediction model achieved an overall accuracy score of 77% and a weighted F1 score of 76.3%. As expected, this model's scores are moderately lower, because predicting a class that can be anywhere from 1 to 80 is a more complex problem than predicting sex, which is a simple binary classification. Also, the confusion matrix itself explains the lower scores as compared to the sex prediction: as discussed in the previous chapter, the data set was not completely balanced through all classes, so the ratio of people aged between 21-40 years was disproportionately high compared to other age groups. Summing up the values across the Truth rows, 23.65% of the people in the test data were aged between 1-20, 44.9% were between 21-40, 20.49% were between 41-60 and only 10.96% were between 61-80 year old. As a result, the model is better at predicting younger people's age, and it fails more often at predicting older ages. Moreover, possibly due to the fact that almost half the people in the dataset were between 21-40 years of age, the model often makes the mistake of predicting this age group even for 1-21 and 41-60 year age ranges, too.

Finally, Table III shows the confusion matrix regarding the race prediction model's performance on the test data.

The model achieved an accuracy score of 83.4%, and a weighted F1 score of 88.9%). Based on this, we can conclude that all the models achieve a considerable accuracy in the predictions. An interesting pattern to note is that the model makes more errors with people in the white race: the most common mistake the model makes is predicting indian, asian and black people to be white.

Summing up the results we can see that sex prediction works the best with 91.8% accuracy, better than the race prediction model's 83.4% accuracy which outperforms the age prediction model's 77% accuracy. While the age prediction model is not as good as the other two models, it still reaches an accuracy that can be dangerous from a privacy standpoint. However, the main takeaway is that the three demographics attributes can be used to re-identify people from face embeddings.

TABLE I
CONFUSION MATRIX OF THE SEX PREDICTION MODEL (ACCURACY=91.8%, RECALL=92.8%, PRECISION=90.9%)

| Truth | | Predicted Sex: *Male* | Predicted Sex: *Female* |
|---|---|---|---|
| | *Male* | 45.54% | 4.65% |
| | *Female* | 3.59% | 46.22% |

TABLE II
CONFUSION MATRIX OF THE AGE PREDICTION MODEL (ACCURACY=77%, RECALL=77%, PRECISION=77.8%)

| Truth | | Predicted Age: *1-20* | Predicted Age: *21-40* | Predicted Age: *41-60* | Predicted Age: *61-80* |
|---|---|---|---|---|---|
| | *1-20* | 17.63% | 5.89% | 0.04% | 0.09% |
| | *21-40* | 0.30% | 42.17% | 2.26% | 0.17% |
| | *41-60* | 0.04% | 6.96% | 11.44% | 2.05% |
| | *61-80* | 0.04% | 1.02% | 4.18% | 5.72% |

## VI. EMBEDDING RE-IDENTIFICATION BY PREDICTING DEMOGRAPHICS

With the three Random Forest Classifier models trained, we were equipped to simulate a re-identification attack using face embeddings against a synthetic database. We carried out attack simulations against databases sizes of 10, 50 and 100 people, which are plausible database sizes for small or medium sized companies.

To construct the synthetic databases with realistic demographics data, we relied on census data from the University of California's Adult Data Set for Machine Learning [34]. This dataset contains over 30,000 records of different types of people including their demographic data (age, sex and race) and the ratio of people believed to be represented by every record. We used the latter weights to sample this dataset to build the smaller databases of 10, 50 and 100. For every person in each database, we then associated photos from the UTKFace dataset [33] that matched their age, race and sex, and used [32] to extract the corresponding facial embeddings from these photos, while taking care not to ever re-use photos that were part of the training data set. In order to suppress any potential bias coming from the randomness, we repeated each experiment with a new synthesized dataset 50 times.

Next, we used our models to predict the sex, age (in 20 year ranges) and race from each embedding, and tried to match the prediction results to people in the original database. By comparing matched records to their corresponding ones in the original database (the ground truth), we could find out how many people's demographic information were correctly

TABLE III
CONFUSION MATRIX OF THE RACE PREDICTION MODEL (ACCURACY=83.4%, RECALL=83.4%, PRECISION=95.2%)

| Truth | | Predicted Race: *White* | Predicted Race: *Black* | Predicted Race: *Asian* | Predicted Race: *Indian* |
|---|---|---|---|---|---|
| | *White* | 24.46% | 0.47% | 0.13% | 0.60% |
| | *Black* | 3.03% | 21.55% | 0.04% | 0.55% |
| | *Asian* | 2.60% | 0.17% | 22.28% | 0.17% |
| | *Indian* | 4.61% | 0.43% | 0.38% | 18.52% |

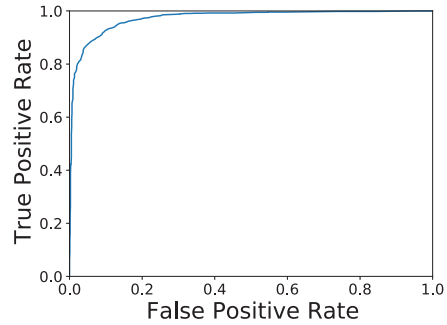



Fig. 2. The ROC curve for the sex prediction model (AUC=97.6%)

how many people's demographic information were correctly predicted. Also, as explained in Section IV, the smaller the size of a person's equivalence class is, the higher their risk of re-identification. So to consider the risks involved with these attacks, we measured the ratio of people falling in equivalence classes of different sizes (1, 2-5, 6-10, 11-20 and 20+). As stated above, we repeated this process 50 times for each smaller database to get an averaged out result.

Figure 3 shows our findings regarding equivalence class sizes. The most successful attacks can be carried out against the smallest database of 10 people, where 16% of all records fall in a unique equivalence class and are thus re-identified, and an additional 33.4% of records fall in an equivalence class of size 2-5, which still means considerable privacy risks. The risks are present even in the case of the databases of size 50 and 100, where the ratio of people falling in a unique equivalence class is 2.36% and 0.98% respectively, and the ratio of people falling in an equivalence class of size 2-5 is 12.72% and 7.18% respectively.

There is a considerable risk of re-identification for many people in all three database sizes simulated. If someone was unique, then we considered that as a successful re-identification. For the rest, the success of re-identification is proportional to the equivalence class size. We used the following metric to determine the overall risk of re-identification in each database size. If we consider the size of an equivalence class to be $k$, and the percentage of people that fall in that equivalence class based on the prediction is $P$, then the re-identification risk of that equivalence class is $P/k$. To get the expected proportion of people re-identified, one has to sum these values for all equivalence classes. In our experiments, these values were 28.90% for the database of 10, 10.38% for the database of 50, and 6.04% for the database of 100 people.

In conclusion, these results show that carrying out re-identification attacks by using face embeddings is indeed possible. Although as there are more people in the database, success of the attack degrades, chances of re-identification are never negligible.

## VII. CONCLUSION

In this paper we discussed potential privacy and security risks associated with the widespread usage of facial recognition technologies, in particular the risks associated with pro-

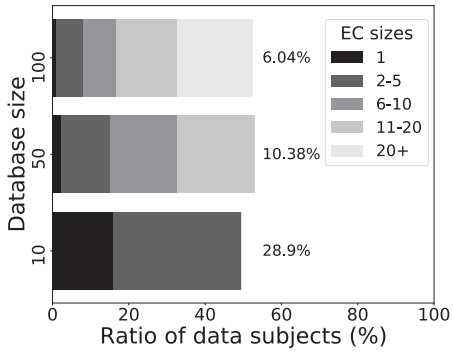



Fig. 3. The ratio of equivalence classes (EC) in the predicted databases ($D$) for various database sizes. Values in parentheses show the expected proportion of re-identified users.

cessing the concerned biometric identifiers. More specifically, we focused on attack that aim to re-identify facial embeddings based on using face embeddings to find out three key pieces of demographics data about the data subjects.

Our goal was to examine the level of accuracy achievable in predicting the sex, age and race from a face embedding. We used a publicly available facial database labeled with these demographic attributes to build a labeled training and testing dataset, and we trained a Random Forest Classifier to predict the sex, age and race from the embeddings.

Based on our findings, it is indeed possible to correctly predict someone's sex, age (within a 20 year range) and race from a face embedding with high accuracies: our models achieved a 90.9% accuracy score on sex prediction, a 83.4% accuracy score on race prediction and a 77% accuracy score on age prediction. As a result, we can consider our theory proven and state that the storing and processing of unprotected face embeddings pose considerable privacy risks as far as re-identification attacks and sensitive data leakage are concerned.

As the final conclusion, we state that further research is necessary to come up with privacy preserving ways to protect embeddings. One idea is to modify the face embeddings in such a way as to keep their utility (e.g. embeddings of the same person should remain close to each other in the vector space after the modification) while protecting them against reverse engineering attacks to make inference more difficult.

## ACKNOWLEDGMENT

The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications).

Project no. FIEK_16-1-2016-0007 has been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary, financed under the Centre for Higher Education and Industrial Cooperation - Research infrastructure development (FIEK_16) funding scheme.

Icons made by Pixel perfect, fjstudio, Freepik, Pause08, surang, Smashicons from www.flaticon.com.

## REFERENCES

[1] L. Introna and H. Nissenbaum, "Facial recognition technology: a survey of policy and implementation issues," 2010.

[2] C. Castelluccia and D. Le Métayer Inria, "Impact analysis of facial recognition," Feb. 2020, working paper or preprint.

[3] "Facial recognition: for a debate living up to the challenges," 2019.

[4] J. Goldenfein, "Facial recognition is only the beginning," 2020.

[5] E. . E. D. P. Board, "Guidelines 3/2019 on processing of personal data through video devices," 2019.

[6] T. Macaulay, "Automated facial recognition breaches gdpr, says eu digital chief," 2020.

[7] S. Stolton, "Leak: Commission considers facial recognition ban in ai 'white paper'," 2020.

[8] E. Commission, "White paper on artificial intelligence: a european approach to excellence and trust," Tech. Rep., 02 2020.

[9] T. Macaulay, "Washington state passes microsoft-approved facial recognition laws," 2020.

[10] D. Lee, "San francisco is first us city to ban facial recognition," 2019.

[11] FRA, "Facial recognition technology: fundamental rights considerations in the context of law enforcement," 2019.

[12] M. Kaya and H. Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, p. 1066, 08 2019. [Online]. Available: DOI: 10.3390/sym11091066

[13] G. Mai, K. Cao, P. C. Yuen, and A. K. Jain, "On the reconstruction of face images from deep face templates," p. 1188–1202, May 2019. [Online]. Available: DOI: 10.1109/TPAMI.2018.2827389

[14] M. Gomez-Barrero and J. Galbally, "Reversing the irreversible: A survey on inverse biometrics," *Computers & Security*, vol. 90, p. 101700, 2020. [Online]. Available: DOI: 10.1016/j.cose.2019.101700

[15] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," *Josa* a, vol. 4, no. 3, pp. 519–524, 1987. [Online]. Available: DOI: 10.1364/josaa.4.000519

[16] M. Turk and A. Pentland, "Eigenfaces for recognition," Journal of cognitive neuroscience, vol. 3, no. 1, pp. 71–86, 1991.

[17] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778. [Online]. Available: DOI: 10.1109/cvpr.2016.90

[19] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[20] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun 2015. [Online]. Available: DOI: 10.1109/CVPR.2015.7298682

[21] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in In NIPS. MIT Press, 2006.

[22] E. Parliament and of the Council, "Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation)," 2016.

[23] E. . E. P. R. Service, "Understanding algorithmic decision-making: Opportunities and challenges," 2019.

[24] X. Dong, K. Wong, Z. Jin, and J.-L. Dugelay, "A cancellable face template scheme based on nonlinear multi-dimension spectral hashing," Cancun, MEXICO, 05 2019. [Online]. Available: DOI: 10.1109/iwbf.2019.8739179

[25] L. Sweeney, "Simple demographics often identify people uniquely," 2000, Working paper.

[26] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Proc. of the 29th IEEE Symposium on Security and Privacy*. IEEE Computer Society, May 2008, pp. 111–125. [Online]. Available: DOI: 10.1109/SP.2008.33

[27] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *2009 30th IEEE Symposium on Security and Privacy*, 2009, pp. 173–187. [Online]. Available: DOI: 10.1109/sp.2009.22

[28] Y.-A. de Montjoye, L. Radaelli, V. K. Singh, and A. ". Pentland, "Unique in the shopping mall: On the reidentifiability of credit card metadata," *Science*, vol. 347, no. 6221, pp. 536–539, 2015. [Online]. Available: DOI: 10.1126/science.1256297

[29] T. Oliphant, "NumPy: A guide to NumPy," USA: Trelgol Publishing, 2006.

[30] W. McKinney, "Data structures for statistical computing in python," in *Proceedings of the 9th Python in Science Conference*, vol. 445. Austin, TX, 2010, pp. 51–56. [Online]. Available: DOI: 10.25080/majora-92bf1922-00a

[31] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg et al., "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[32] A. Geitgey, "face recognition: The world's simplest facial recognition api for python and the command line," 2020.

[33] Y. Zhang Zhifei, Song and H. Qi, "Age progression/regression by conditional adversarial autoencoder," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2017. [Online]. Available: DOI: 10.1109/cvpr.2017.463

[34] D. Dua and C. Graff, "UCI machine learning repository," 2017.

**István Fábián** is a technical assistant at the Budapest University of Technology and Economics (BME) since 2019. He is a member of the Balatonfüred Student Research Group. His research interests include privacy and security in machine learning, and he is also working on projects related to IoT and Industry 4.0 in the BME Technology Center.

**Gábor György Gulyás** has been involved with Privacy Enhancing Technologies since 2005. In 2015 he obtained the degree of PhD Budapest University of Technology and Economics (BME). The focus of his thesis was on how privacy and anonymity could be preserved in social networks against largescale re-identification attacks. Between 2015 and 2018 he was a PostDoc and Research Engineer in the Privatics team at INRIA (France). There, he was working on research projects related to web privacy and at the intersection of machine learning and privacy. Since 2019 he is a research fellow at BME with a special focus on (but not limited to) the privacy issues related to the IoT and machine learning technologies.

# Adapting IT Algorithms and Protocols to an Intelligent Urban Traffic Control

Levente Alekszejenkó[1] and Tadeusz Dobrowiecki[2]

***Abstract*—Autonomous vehicles, communicating with each other and with the urban infrastructure as well, open opportunity to introduce new, complex and effective behaviours to the intelligent traffic systems. Such systems can be perceived quite naturally as hierarchically built intelligent multi-agent systems, with the decision making based upon well-defined and profoundly tested mathematical algorithms, borrowed e.g. from the field of information technology.**

**In this article, two examples of how to adapt such algorithms to the intelligent urban traffic are presented. Since the optimal and fair timing of the traffic lights is crucial in the traffic control, we show how a simple Round-Robin scheduler and Minimal Destination Distance First scheduling (adaptation of the theoretically optimal Shortest Job First scheduler) were implemented and tested for traffic light control. Another example is the mitigation of the congested traffic using the analogy of the Explicit Congestion Notification (ECN) protocol of the computer networks. We show that the optimal scheduling based traffic light control can handle roughly the same complexity of the traffic as the traditional light programs in the nominal case. However, in extraordinary and especially fastly evolving situations, the intelligent solutions can clearly outperform the traditional ones. The ECN based method can successfully limit the traffic flowing through bounded areas. That way the number of passing-through vehicles in e.g. residential areas may be reduced, making them more comfortable congestion-free zones in a city.**

***Index Terms*—intelligent traffic control, connected vehicles, congestion notification, Intelligent Transportation Systems (ITS), Intelligent Traffic Light System (ITLS)**

## I. Introduction

As our vehicles become more and more sophisticated (up to being self-driving and autonomous, *smart cars* for convenience) and the traffic infrastructure itself also evolves, communication between smart cars (V2V), or between smart cars and various parts of the infrastructure (V2I), or even between various elements of the infrastructure (intersections, parking lots, etc.) is no longer a fiction. If the infrastructure and the smart cars are also capable of cooperative actions by following the exchanged communication messages, it is possible to form intelligent multi-agent systems to improve road safety, reduce traveling times, costs and pollution, or even to mitigate congestion as well. For more details, see Section III.

However, the internal behavior (the decision making) of these agents has to be defined. Among others such agents have to calculate answers to the e.g. following questions: *Would it be beneficial for a smart car to join a group of cars in front of it? When should an intelligent traffic light provide a green-light for a particular platoon of smart cars? When shall an intelligent traffic light ask one of its neighbor junctions to reduce its output to prevent congestion?* To be able to answer these questions, Round-Robin, Minimal Destination Distance First, and Explicit Congestion Notification protocols are proposed in Section IV. When we defined these methods, we had the presumption that every vehicle in the traffic are autonomous and can communicate with each other.

Besides integrating various components of an intelligent transportation system into a hierarchical multi-agent system, adopting the aforementioned protocols to the road traffic domain, especially the ECN protocol, is the principal novelty in our research. The proposed solutions were also tested by simulations of different (hopefully realistic) scenarios, using the Eclipse SUMO microscopic traffic simulator tool [1]. The measurements and their results are summarized in Section V.

## II. Literature Review

As the first coordinated traffic lights were created more than one hundred years ago [2], the literature of traffic control contains many interesting articles, books, and lecture-notes. Even though this is a well-researched area, perhaps the major problem of transportation, the congestion, still exists.

Traffic signal coordination, green-waves, are nowadays mainly created by methods depending on analyzing statistical data, like TRANSYT and SCOOT [3]. Since those algorithms were created decades ago, they might not be able to handle the problems of today's traffic. Thus, it may be helpful to implement new, intelligent methods into the traffic lights. One of these approaches may be the usage of Minimal Destination Distance First [4] control which is analogous to the theoretically optimal scheduling algorithm, called the shortest job first. Unfortunately, this method is unfair on its own, therefore it shall be modified to use it in real-life [5].

It is natural to treat the participants of urban traffic (e.g. vehicles, infrastructural elements, traffic lights, etc.) as a multi-agent system. In this framework, novel ideas can also be experimented with, such as a time-slot booking to pass through at the intersections, explained in [6]. Unfortunately, there is no guarantee that a smart vehicle will arrive on-time to a certain intersection, but this method contains the possibility to withdraw the already posted bookings. The problem is that the state-space of such a system can be enormous, therefore this

[1,2] Balatonfüred Student Research Group
[2] Budapest University of Technology and Economics, Budapest, Hungary
(e-mail: ale.levente@gmail.com and tade@mit.bme.hu)

DOI: 10.36244/ICJ.2020.2.8

and similar algorithms require a vast amount of computational time and memory space.

Method of significantly lower complexity is proposed in [7]. This proposal varies the phase time of traffic lights like the SCOOT method does, but this method varies the phase times of multiple traffic lights at the same time. Therefore it creates arterial directions. As our research showed [5], the main advantage of intelligent traffic control is that it behaves better in extraordinary situations. In the investigated cases, particular arterial directions were closed, and secondary routes opened to obligatory use, due to the road closures. Therefore in our research, we tried to avoid creating arterial directions.

In a grid-like road network, for example, typical to the U.S., there are at least two routes with the same cost between any two points of the road network. [8] takes advantage of this fact, optimizing traffic both in time and space in over-saturated scenarios. Unfortunately, this method cannot be simply applied in irregularly shaped road networks, prevalent e.g. in Europe.

Traffic flow can also be described with the concepts borrowed from Economics. Therefore some economical formulas and methods also can be applied in the domain of road traffic. [9] presents an economical approach to optimize the flow of traffic. However, it is not a true real-time solution, since the phases shall be recalculated always when a new car approaches an intersection. Therefore this method is also really of high computational complexity.

Computationally, a much simpler approach is to create individual agents at traffic lights and design an algorithm or a physical phenomenon which automatically provides the signal coordination. [10] shows that traffic coordination might easily be implemented by actuated traffic lights. In this case, communication between traffic light is not necessary, since the incoming platoons of vehicles can synchronize those intersection managers when they arrive at the corresponding induction loop detectors. [11] also suggest using a distributed traffic light control, in which controller agents can play an evolutionary game. By playing the game individually, the agents might be able to find the globally optimal solution as well. Unfortunately, this work does not mention what happens when the system is adapting to the recently changed traffic. There is a possibility that an almost endless traffic congestion forms in this transient state (considering that both the traffic itself and the traffic controllers are in a transient state for a while).

For this reason, our ECN-based solution, presented in this paper, is a much more conservative one. It is also a distributed solution, but a limited amount of information is shared among the topologically neighboring intelligent traffic controllers. Based on this information, our method solves a relatively small optimization (integer programming) problem. [12] attests that sharing information with neighboring intersection managers can be beneficial for targeting the globally optimal solution. Based on these results, optimizing the scheduling of more intersection controllers (e.g. for a dedicated direction, [13]) at the same time might not be worth the increased computational time and complexity.

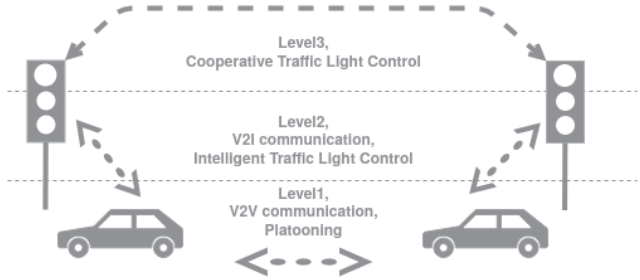



Fig. 1. Example of a three-layered, intelligent multi-agent system of urban traffic.

## III. URBAN TRAFFIC AS AN INTELLIGENT MULTI-AGENT SYSTEM

Autonomous vehicles, smart traffic lights are intelligent agents on their own. By using their capability of communication, cooperative, multi-agent systems can be formed.[3] In the following, we assume these agents to be trustworthy and bonafide, cooperative, and being able to perform actions prescribed by the defined protocols. The communication itself is free of lost packets, the bandwidth is enough to transmit all the messages and the delay of the transmission does not have any effect on the agents' behavior. On this basis, we can identify three layers of the cooperating agents and the related intelligent behavior.

*In the first layer* (lowest, vehicle-level, see Figure 1), inter-vehicular communication is used to form groups of smart cars, the so-called platoons. Vehicles in a platoon can keep shorter following distances and can perform some maneuvers together, like e.g. changing lanes. When multiple vehicles change lanes together, they might have a smaller impact on the flow of the traffic, compared to changing lanes individually. Unfortunately, in an urban environment, there is usually not enough space and time to perform complex maneuvers, therefore we believe only simple methods can be executed there. Thus, platoons in urban scenarios are expected to form in ad-hoc ways, in smaller groups, and will have a relatively shorter lifetime.

Besides using simple platoon movements, there are many other ways to improve traffic flow and to reduce congestion in modern cities. For example, smart cars can inform the intelligent traffic lights (V2I communication) about their approaching. Based on this, traffic lights can attempt to compute an optimal signal plan according to the actual traffic demand. This will be *the second layer* (vehicle-to-intersection) of the analyzed multi-agent system.

As *the third layer*, we can assume that the traffic lights also communicate with each other in an attempt to limit the formation of congestion in a wider geographical area. Basically, congestion forms when more vehicles do arrive at

[3]Trams and trains can easily be treated as autonomous vehicles on their own. Even pedestrians can be part of this concept, as they can place their demands by pushing a button at intersections, or they can be detected by simple photocells. As they can be informed by traditional lights, theoretically their presence (if orderly) is indifferent to the autonomous vehicles. The difference is simply technical as orderly behaving pedestrians differ from autonomous vehicles only in sensing and signaling.

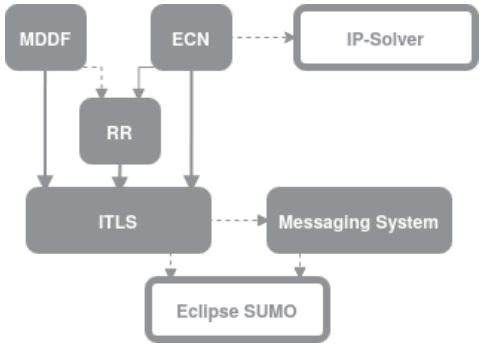



Fig. 2. Functional overview of the proposed system, implemented by extending the Eclipse SUMO microscopic traffic simulator tool. The main functional blocks include a communication module (Messaging System) and the different kinds of ITLS. RR is an ITLS technique on its own, but it is also used by the ECN-method, and contained by the MDDF method as well. ECN method uses an external IP-Solver package.

an intersection than that intersection throughput capacity. If the number of incoming vehicles could be somehow limited, then the congestion could be avoided.

## IV. Agent Algorithms and Protocols

The intelligent multi-agent system delineated in Section III provides a framework in which the actual behavior of particular agents is yet to be defined. In the following, we present some possible algorithms governing the *second and the third layer* of the system.[4] We will mainly focus on various algorithms suitable for the intelligent traffic light controllers (ITLS), as these agents participate in both higher system layers.

Considering the second layer, ITLS are only responsible for controlling a single intersection. We will call this an *individual scheduling*. In the third layer, however, the ITLS share information, therefore we will call it a *cooperative scheduling*. Actions of a cooperative scheduler control the "lamps" of the scheduler's intersection while these actions are based on both on the scheduler's perceptions and the information received from the topologically neighboring schedulers. For a functional overview, see Figure 2.

### A. Simple Individual Scheduling: Round-Robin Protocol

Individual scheduling, in our solution, is based on well-known algorithms of scheduling theory. One of the simplest scheduling algorithms is the Round-Robin scheduler (RR). RR provides green-light for every direction periodically. A *preemptive* version was implemented, which means that green-light periods can be shorter or even skipped if there are no more smart cars to pass through in a specific direction. In traffic engineering, this is called *phase-skipping*. In addition to its simplicity, RR provides fair scheduling for all traffic directions.

[4]Complex actions in the first layer (i.e. individual vehicles and platoons) also might be defined, but since the time and space is limited for difficult maneuvers in urban scenarios, we do not discuss this possibility here.

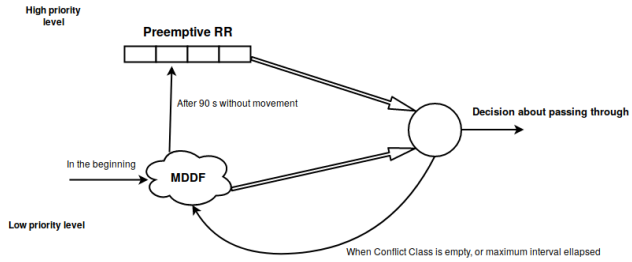



Fig. 3. Functional overview of the MDDF ITLS.

### B. Complex Individual Scheduling: MDDF

In principle, there exists an optimal scheduling algorithm, the so-called Shortest Job First scheduler. In the case of urban traffic control, it is a good question what the "shortest job" should mean. One possibility is to pick a vehicle, which is closest to its destination[5]. The idea behind this is that a shorter distance has a smaller impact on the traffic infrastructure, therefore our "job is shorter" too. Let us call it the *Minimal Destination Distance First (MDDF)* scheduling. However, the MDDF scheduling is not fair in itself. Assume, a lonely car is waiting at an intersection to pass, for example, being at the beginning of a route to a very distant destination. Yet vehicles with significantly closer destinations are continuously arriving. The car with the faraway destination can thus wait forever without getting through this intersection. Assuming that the shortest distances are distributed uniformly between every possible route through an intersection, it is a rare but still problematic case.

If MDDF is combined with RR scheduler [5], [14], then the protocol will be fair. To make it so, the protocol should be a multilayered scheduler with two priority levels. Every direction will be scheduled by the MDDF scheduler when the vehicles arrive at a particular ITLS. If a limited time (here 90 seconds) elapses without receiving a green light for a particular direction, this direction will be scheduled then by an RR scheduler. The RR is at a higher priority, so if there are any directions which must be scheduled by the RR, they will receive green light before those scheduled by the MDDF, see Figure 3. This way, the scheduler will provide fair scheduling.

### C. Congestion Avoidance in Computer and Road Networks

The idea of synchronizing signals of neighboring intersections is not a new one. Traffic signal coordination, known mainly as *green-waves*, has been applied in traffic engineering since 1917, to help the flow of the traffic. Congestion is, however, not a unique phenomenon to the roads of our cities. Computer networking also faces the problem of congestion. If there are more messages to send than the network can handle in a given amount of time, computer networks also become congested.

[5]To be precise, the resolution of our scheduling solutions is a so-called *conflict-class*. The CAVs of a conflict-class can pass through an intersection simultaneously without the risk of an accident. It is analogous to the traditional traffic lanes which can receive green lights at the same time.

Congestion in computer networking can be mitigated in numerous ways. For example, *exponential backoff* [15] retransmits packets when collisions occur at a point of time, which is selected randomly from an exponentially growing[6] time range. It is a relatively effective method in computer networks, but it cannot be applied to road traffic.

Another method in computer networking is the *sliding window protocol* [16]. Its basic idea is that the number of packets transmitted at the same time shall be limited. It would be theoretically applicable to the traffic as well, but in transportation systems, platoons, groups of cars, show many benefits, therefore mitigating or eliminating them might not be so beneficial at the end. Thus, this method is not in the focus of this article.

To the urban traffic we can apply also the *Explicit Congestion Notification (ECN)* [17] used in the computer networks. The idea behind this algorithm is that the receiver router/intersection informs the sender router/intersection (sends an ECN-signal) when it cannot handle the amount of the incoming messages/vehicles. By catching this notification, the sender is expected to reduce its output until the receiver's further notice.

### D. Cooperative Scheduling: ECN

Implementing the ECN method in the ITLS environment of several intersections is quite a challenging task (see Figure 4). The state-space of such a system can be enormous, therefore storing all the possibilities and searching among them is not necessarily feasible.[7] However, storing all the possible set-ups of the "traffic lights"[8] of an intersection is unavoidable if we want to create a pre-programmed ITLS. This ITLS would use this huge list of set-ups in a kind of a look-up-table, therefore given the current state of the traffic and the incoming congestion notifications, the ITLS would be able to select the next signal-phase by searching this particular look-up-table.

Unfortunately, even in modern embedded systems, such look-up-table based solution is almost impossible to implement. Instead of pre-programming the ITLS, signal-phases can be generated in real-time. The calculation of a simple signal phase is mathematically equivalent to solve an integer programming problem (IP). Since modern and powerful IP-solvers are available, this method can be easily ported even to embedded devices.

The variables of the proposed IP are constrained to $\{0, 1\}$ values. Every direction will have its corresponding variable, which will be 0 if the direction is to receive a red light, and 1 if it is to receive a green.

The constraints attached to the IP prescribe that only non-conflicting directions can go through the intersection (passing through the intersection simultaneously without risking collision). Besides, with constraints, some traffic lights can be specifically set to green or to red as desired. The optimum criterion is trivially to maximize the number of directions that currently receive the green light. This approach also avoids creating arterial directions. The lack of an arterial direction, a "main route" might be beneficial in extraordinary situations, because the congested vehicles can reach an alternative route much more easily [5].

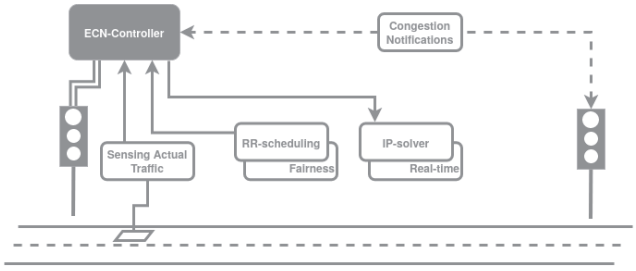



Fig. 4. Overview of an ECN traffic light controller. It is a real-time, fair scheduler. ECN is capable of sensing the actual traffic, and of notifying its neighbors about a forming congestion. When one of its neighbors sends the congestion notification, the ECN-controller will reduce its output accordingly.

When generating a signal phase, at least one direction shall be selected to receive a green light. For this decision, a simple Round-Robin scheduler is used. As it is discussed in Section IV-A, the RR scheduler can provide a fair scheduling[9]. Technically, it means one variable of the IP has to be constrained to 1, regarding the scheduling decision of the RR.

To make signal plans, individual signal phases have to be calculated periodically. Our solution recalculates signal phases when there are no more cars in the direction which currently receives a green light. A 90 s time-limit is also set as the maximum time delay between two recalculations.[10]

One problem is yet to solve. It is necessary to decide when congestion is about to form. Without a clear definition of congestion (there is no accepted unequivocal definition of congestion), it is quite a difficult task. Thus based on preliminary simulations, we simply calculated the traffic density, which can provide the highest traffic flow in a given locality. We accept that there is congestion forming when 90% of this level is reached. The ECN-signal is sent then at this event.

## V. Measurements and Results

The proposed protocols were tested under a suitably extended version of the Eclipse SUMO, an open-source, microscopic traffic simulation program (see Figure 2). The used network was the BAH-intersection[11] of Budapest, together with the wider roads of its neighborhood, see Figure 5.

[6] This time range is proportional to the number of the unsuccessful transmissions.

[7] Comparing to traditional phase-skipping, where the number of possible states is linearly proportional to the $N$ number of intersections, here the individual control of directions is necessary. Given $N$ intersections, the number of directions is of order $N^2$ (calculated as the maximum number of edges in road network graph).

[8] In an intelligent system, it might be a simple message, not necessarily a physically existing traffic light.

[9] RR scheduler, in this case, is also implemented as a preemptive RR scheduler. This helps increase the traffic flow in the currently popular directions. Therefore it creates arterial directions dynamically, in accordance with the actual traffic demand.

[10] It is alike as in the preemptive Round-Robin scheduler.

[11] Intersection of Hegyalja út, Jagelló út, Villányi út, Budaörsi út and Alkotás utca.

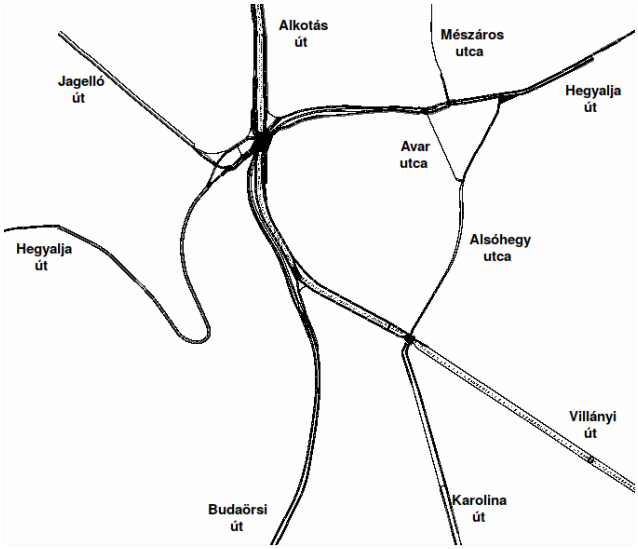



Fig. 5. The simulated network of BAH intersection and its neighborhood.

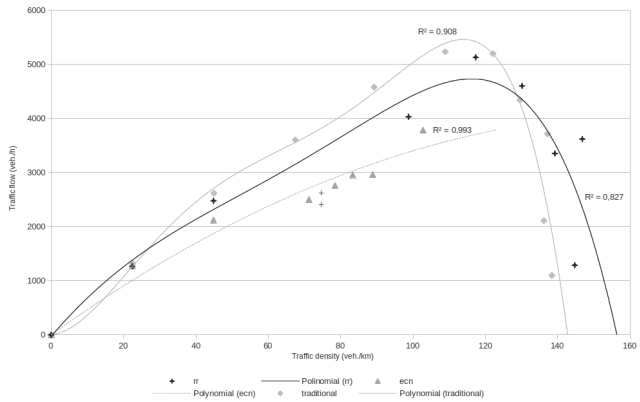



Fig. 6. The results of the simulation, depicted on a Macroscopic Fundamental Diagram of traffic. The values provided by the MDDF ITLS are not shown, because in saturated cases at higher flow of the traffic, they practically coincide with the RR ITLS.

Regular traffic demands were fed into the simulator (eg. night traffic, morning traffic, noon traffic) as well as some irregular traffic (Budaörsi út is closed) scenarios[12].

We can see from the results (Table I and Table II), that considering the average waiting time (for example at red lights) and the average traveling time, these indicators are reduced when the proposed intelligent protocols are utilized for irregular traffic situations. On the other hand, in the regular traffic situation, the signal program of the traditional control system (possibly optimized for such regular demands) performs very well (see Figure 6) and the intelligent protocols leave a little margin to the improvement.[13]

TABLE I
SIMULATION RESULTS OF "IRREGULAR1" CASE

| Test case | Arrived (%) | Waiting Time (s) | Average Traveling Time (s) |
|---|---|---|---|
| Traditional | 33.81 | 29.68 | 170.55 |
| RR | 29.19 | 12.117 | 174.87 |
| MDDF | 22.77 | 12.41 | 154.02 |

TABLE II
SIMULATION RESULTS OF "IRREGULAR2" CASE

| Test case | Arrived (%) | Waiting Time (s) | Average Traveling Time (s) |
|---|---|---|---|
| Traditional | 38.48 | 36.44 | 199.38 |
| RR | 32.71 | 11.43 | 170.07 |
| MDDF | 34.39 | 10.74 | 176.72 |

The cooperative scheduling was also tried in a regular morning scenario. The BAH-intersection and some of its algorithmically selected neighbors [18] were programmed to work according to the ECN-algorithm. This algorithm selects a group of neighboring intersections which are coupled to each other. The coupling of the neighboring intersections means a traffic path constraint, i.e. if the vehicles pass one of these intersections, they are constrained to pass the other one also. By running this algorithm, we can identify that the junction of "Jagelló út" with "Hegyalja út", and the junction of "Villányi út" and "Budaörsi út" are coupled to the BAH intersection. Thus, they are governed by the ECN ITLS. Every other ITLS in this system is signaled by simple RR ITLS.

As theoretically expected and visible on the so-called Macroscopic Fundamental Diagram (MFD, see Figure 6), this method limits the flow of the traffic to a certain level, significantly smaller than the maximum achievable flow in this region. Thus, this method cannot be applied to elevate the aggregated number or speed of the vehicles in the simulated scenario. However, in the ECN-coordinated cases, the MFD lacks the descending (jammed traffic) branch of the diagram, therefore the traffic system remains in a stable state.

This proposed method, however, is capable of mitigating congestion in bounded zones of the city traffic [18], especially where the limitation of traffic is desired. Such areas are the residential zones, areas around parks and other recreational facilities, surroundings of hospitals, etc. The major advantage of using an ECN ITLS in such areas, compared to the classical static methods, is that ECN-based control means no inconvenience for the inhabitants and their visitors. On the contrary, commuters cutting-through can be easily banned, as the roads in these zones will not become beneficial alternatives for them.

## VI. CONCLUSION

Representing road traffic as a cooperative intelligent multi-agent system provides a framework for modeling intelligent vehicles and infrastructural elements of the cities of the future. To analyze the possible behaviors of the various modeled parts of the road network, the decision making capabilities of the agents have to be defined, best based on the well-tested or mathematically precisely known methods.

[12] *Irregular1* case. The obstacle is northbound of "Budaörsi út", can be bypassed via Karolina and Villányi streets.
*Irregular2* case: Obstacle is southbound of "Budaörsi út", the 0bypass route is via "Hegyalja út".

[13] The traffic flow is a commonly calculated value. It is the product of the traffic density ($\frac{\text{vehicles}}{km}$) and the mean velocity of the vehicles ($\frac{km}{h}$). These values can be measured by different types of detectors, cameras, etc.

In this paper, we borrowed ideas from computer networks and scheduling theory to create intelligent traffic light controller algorithms. The intelligent single intersection schedulers perform similarly to the traditional control systems in normal traffic conditions. However, in extraordinary situations, intelligent traffic light control can outperform the traditional one. Cooperative scheduling, based on the ECN algorithm and concerting the activity of several intersections, can reduce traffic flow in the whole area. It can be certainly beneficial in some cases, but this method can also avoid reaching the downgrading of the traffic flow.

As the next step of the research, it would be extremely beneficial to investigate whether it would be feasible to set the flow limitation of the ECN-based control to the desired level. If possible, by setting this limit to just below the maximum flow value, we will, therefore, be able to avoid the congestion at a minimal limitation to the maximum achievable flow. Thus, we would be able to keep the traffic flowing close to the theoretical maximum throughput of a given road network.

## ACKNOWLEDGMENT

The research has been supported in part by the BME – Artificial Intelligence FIKP grant of EMMI (BME FIKP-MI/SC) and in part by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications). The results presented in the research report were established in the framework of the professional community of Balatonfüred Student Research Group of BME-VIK to promote the economic development of the region. During the development of the achievements, we took into consideration the goals set by the Balatonfüred System Science Innovation Cluster and the plans of the "BME Balatonfüred Knowledge Center", supported by EFOP 4.2.1-16-2017-00021.

## REFERENCES

[1] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018, [Online]. Available: https://elib.dlr.de/124092/, DOI: 10.1109/ITSC.2018.8569938.

[2] M. G. Lay, *Ways of the World: A History of the World's Roads and of the Vehicles That Used Them*. Rutgers University Press, 1992, isbn: 9780813526911.

[3] D. I. Robertson, "Research on the transyt and scoot methods of signal coordination," vol. 56, no. 1, pp. 36–40, 1986, iSSN: 0162-8178.

[4] F. Ahmad, S. A. Mahmud, G. M. Khan, and F. Z. Yousaf, "Shortest remaining processing time based schedulers for reduction of traffic congestion," in *2013 International Conference on Connected Vehicles and Expo (ICCVE)*, Las Vegas, NV, USA, 2013, pp. 271–276, DOI: 10.1109/ICCVE.2013.6799805.

[5] L. Alekszejenkó and T. Dobrowiecki, "Intelligent vehicles in urban traffic – communication based cooperation," *The IEEE 17th World Symp. on Applied Machine Intell. and Inform. (SAMI 2019), Herl'any, Slovakia*, Jan 2019, DOI: 10.1109/SAMI.2019.8782778.

[6] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," *Journal of Artificial Intelligence Research*, vol. 31, pp. 591–653, 2008, DOI: 10.1613/jair.2502.

[7] S. Nigarnajagoool and H. Dia, "A multi-agent approach to real-time traffic signal optimisation," in *29th Australasian Transport Reserach Forum, Surfers Paradise, Queensland, Australia*, September 2006, isbn: 1 877040 56 8.

[8] M. Girianna and R. Benekohal, "Dynamic signal coordination for networks with oversaturated intersections," *Transportation Research Record*, vol. 1811, pp. 122–130, 01 2002, DOI: 10.3141/1811-15.

[9] A. Deligkas, E. Karpas, R. Lavi, and R. Smorodinsky, "Traffic light scheduling, value of time, and incentives," in *Proc. of the Twenty-Seventh International Joint Conference on Artificial Intelligence Main track.*, July 2018, pp. 4743–4749, DOI: 10.24963/ijcai.2018/659.

[10] P. Wagner, R. Alms, J. Erdmann, and Y.-P. Flötteröd, "Remarks on traffic signal coordination," in *EPiC Series in Computing*, vol. 62, 08 2019, pp. 244–255, DOI: 10.29007/flbm.

[11] A. Bazzan, "A distributed approach for coordination of traffic signal agents," vol. 10, pp. 131–164, 2005, DOI: 10.1007/s10458-004-6975-9

[12] L. Kuyer, S. Whiteson, B. Bakker, and N. Vlassis, "Multiagent reinforcement learning for urban traffic control using coordination graphs," in *Machine Learning and Knowledge Discovery in Databases*, W. Daelemans, B. Goethals, and K. Morik, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 656–671, DOI: 10.1007/978-3-540-87479-9_61.

[13] B. Beak, "Systematic analysis and integrated optimization of traffic signal control systems in a connected vehicle environment," 2017. [Online]. Available: http://hdl.handle.net/10150/626304

[14] L. Alekszejenkó and T. P. Dobrowiecki, "Sumo based platform for cooperative intelligent automotive agents," in *SUMO User Conference 2019, ser. EPiC Series in Computing*, M. Weber, L. Bieker-Walz, R. Hilbrich, and M. Behrisch, Eds., vol. 62. EasyChair, 2019, pp. 107–123, [Online]. Available: https://easychair.org/publications/paper/Vp8w, DOI: 10.29007/sc13.

[15] "Ieee standard for ethernet," IEEE Std 802.3-2015 (*Revision of IEEE Std 802.3-2012*), pp. 1–4017, 2016, isbn: 978-0-7381-9626-8.

[16] V. Jacobson, R. Braden, and D. Borman, "Tcp extensions for high performance, rfc 1323." The Internet Society, Network Working Group, May 1992, [Online]. Available: https://www.ietf.org/rfc/rfc1323.txt.

[17] S. Floyd, K. K. Ramakrishnan, and D. L. Black, "The addition of explicit congestion notification (ecn) to ip, rfc 3168." The Internet Society, Network Working Group, September 2001, [Online]. Available: https://tools.ietf.org/html/rfc3168.

[18] L. Alekszejenkó and T. P. Dobrowiecki, "Alleviating congestion in restricted urban areas with cooperative intersection management," in IntelliSys 2020, September 2020, (Accepted for publication in Springer series "Advances in Intelligent Systems and Computing").

**Levente Alekszejenkó** was born in Fehérgyarmat, Hungary, in 1996. He is a member of the Balatonfüred Student Research Group. He received the B.Sc. degree in computer science engineering from Budapest University of Technology and Economics, Budapest, Hungary in 2019. His M.Sc. studies in computer science engineering are currently ongoing at the Budapest University of Technology and Economics, Budapest, Hungary. He is currently working at evopro Innovation Ltd., Budapest, Hungary as a Junior Software Engineer. His research interests include multi-agent intelligent systems, especially their application in intelligent transportation systems and road traffic simulation.

**Tadeusz P. Dobrowiecki** was born in Warsaw, Poland, in 1952. He received the M.Sc. degree in electrical engineering from the Technical University of Budapest, Budapest, Hungary, in 1975, the Ph.D. (candidate of sciences) degree in 1981, and the DSc (Doctor of Sciences) degree in 2018, both from the Hungarian Academy of Sciences.

After spending one year as a Professional System Engineer, he joined the Department of Measurement and Information Systems, Budapest University of Technology and Economics, Budapest, Hungary, as a Staff Member, where he is currently a Full Professor. He is involved in teaching artificial intelligence and system identification. His current research interests include technical applications of artificial intelligence, and advanced system identification problems.

# Comparison of Non-Linear Filtering Methods for Positron Emission Tomography

Dóra Varnyú[1] and László Szirmay-Kalos[2]

***Abstract*—As a result of the limited radiotracer dose, acquisition time and scanner sensitivity, positron emission tomography (PET) images suffer from high noise. In the current clinical practice, post-reconstruction filtering has become one of the most common noise reduction techniques. However, the range of existing filters is very wide, and choosing the most suitable filter for a given measurement is far from simple. This paper aims to provide assistance in this choice by comparing the most powerful image denoising filters, covering both image quality and execution time. Emphasis is placed on non-linear techniques due to their ability to preserve edges and fine details more accurately than linear filters. The compared methods include the Gaussian, the bilateral, the guided, the anisotropic diffusion and the non-local means filters, which are examined in both static and dynamic PET reconstructions.**

***Index Terms*—positron emission tomography, image denoising, post-reconstruction filtering, gaussian, bilateral, median, anisotropic diffusion, non-local means, guided filter, efop**

## I. INTRODUCTION

POSITRON emission tomography (PET) is an imaging technique used to observe biochemical or pharmacological processes in the body. As it provides functional information, PET is a particularly helpful tool for early diagnosis of diseases and pharmakinetic studies. However, the applicable radioactive dose and the acquisition time are severely limited (cost, physiological effect) and the sensitivity of the imaging system is also generally low. Because of this, PET images suffer from high levels of noise, which can make small lesions such as early-stage tumors impossible to spot.

One possible way to suppress noise is to introduce a penalty term into the maximum-likelihood optimization [1]. However, determining the appropriate parameter values is challenging because they depend on the measured data and the reconstruction settings. Moreover, if the penalty function is not convex, optimization becomes complex and resource-intensive [2].

Another possible solution for noise reduction is early termination [3], which involves ending the iterative reconstruction algorithm well before its convergence. The determination of a stopping point is quite challenging and usually a compromise has to be made between image detailedness and noisiness.

This study focuses on post-reconstruction filtering for image denoising. There are various filters in current clinical application with different characteristics and resource requirements. Due to its simplicity, the Gaussian filter [4] is most commonly used, but it smoothes out image structures such as tissue boundaries or small lesions besides the noise, thus can deteriorate important clinical information. Better results may be achieved using non-linear filters, since they better preserve the non-linear features of the image such as edges and boundaries. Moreover, certain types of noise can only be effectively removed with non-linear filters. A common example is salt-and-pepper noise, against which the median filter is most effective [4]. Another widely used non-linear filter is the bilateral filter [5], [6], which can also be considered as an extension of the Gaussian kernel. However, in certain scenarios, the bilateral filter can introduce false edges in the image (gradient reversal problem) [7]. An alternative that is free of the gradient reversal problem is the guided filter [7], which produces its output using a guidance image. As the guidance can come from another imaging modality (e.g. CT or MRI) [8], guided filtering makes it possible to take into account anatomical tissue boundaries during filtering. The drawback of this filter is that it is challenging to determine the best guidance and parameter settings as there are many options to choose from and they must be tuned to the measured data. In our previous work [9], we have investigated this topic and proposed several promising guidances for both static and dynamic reconstructions.

Another tool that is able to incorporate high-resolution anatomical images to enhance the output of PET is the anisotropic diffusion (AD) filter [10], [11]. However, it often results in artificially piecewise smooth regions [12]. A more recent alternative is the non-local means (NLM) filter [13], which smooths intensities by the weighted average of intensities in a large neighborhood according to their similarities and was used successfully for PET image denoising in various scenarios [12], [14].

The purpose of this paper is to provide a comprehensive comparison of the different filters for PET in terms of the quality of the output image and the runtime of the operation.

The structure is as follows. In Section II, we give a brief overview of the examined filters. Section III analyzes noise reduction in static reconstructions, while dynamic reconstructions are investigated in Section IV. Finally, the paper is closed with conclusions in Section V.

## II. OVERVIEW OF EXAMINED FILTERS

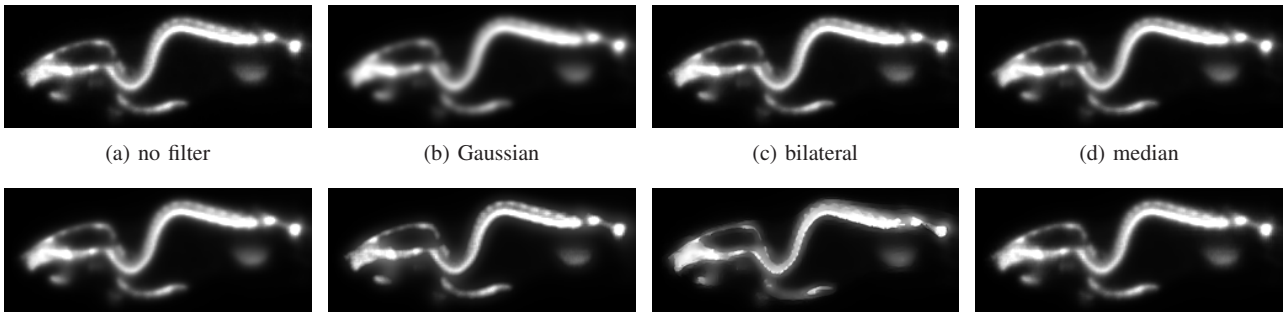
Filtering computes output image $Q$ from input image $P$ by either a linear or a non-linear algorithm. To visually compare the outputs of the examined methods, we performed filtering on a mouse scan measured on Mediso's *nanoScan PET/CT* (Fig. 1) and on a human scan measured on Mediso's *AnyScan human PET/CT* (Fig. 2).

[1] Balatonfüred Student Research Group

[1,2]Department of Control Engineering and Information Technology, Budapest University of Technology and Economics, Budapest, Hungary.

Contacts: varnyu.dora, szirmay@iit.bme.hu

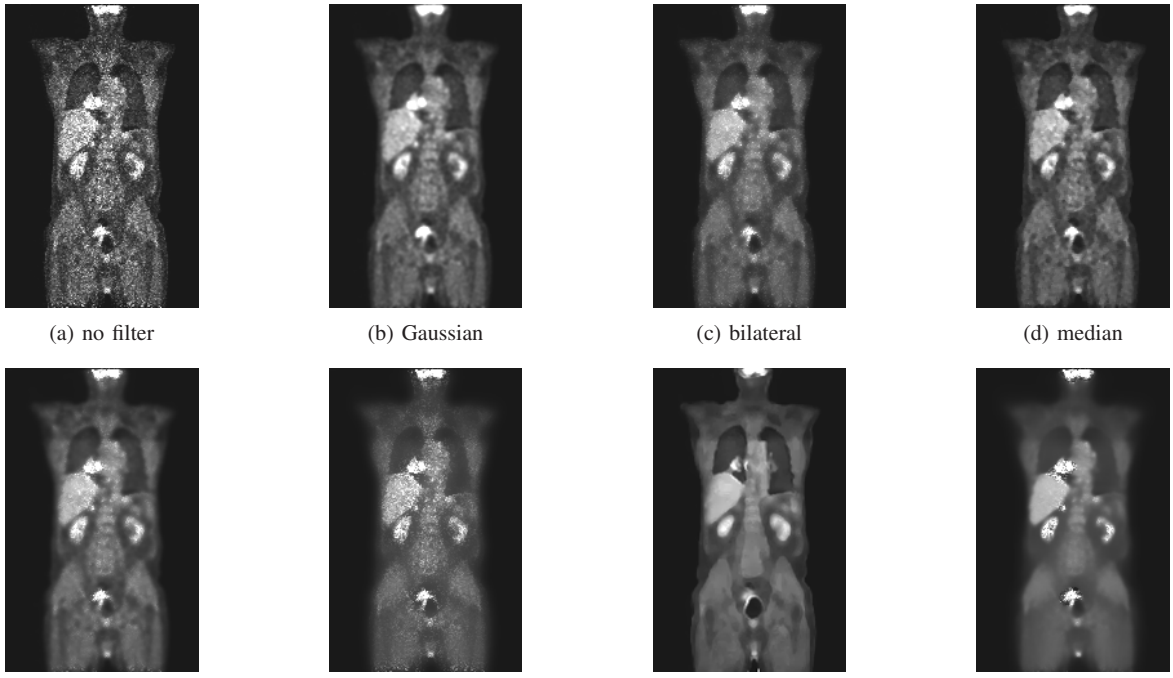
DOI: 10.36244/ICJ.2020.2.9



(a) no filter (b) Gaussian (c) bilateral (d) median

(e) input guided (f) dual-channel guided (g) anisotropic (h) non-local means

Fig. 1: Post-reconstruction filtering of a mouse scan using different filter algorithms



(a) no filter (b) Gaussian (c) bilateral (d) median

(e) input guided (f) dual-channel guided (g) anisotropic (h) non-local means

Fig. 2: Post-reconstruction filtering of a human scan using different filter algorithms

### A. Gaussian Filter

The Gaussian filter replaces voxel activities with the Gaussian-weighted average of the activities of adjacent voxels. That is, it convolves with a filter kernel containing Gaussian weights $g_\sigma(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot \exp(-\frac{x^2}{2\sigma^2})$, where $x$ is an arbitrary real number parameter and $\sigma$ is the standard deviation.

### B. Bilateral Filter

The operation of the bilateral filter is similar to that of the Gaussian filter, however, the weights depend not only on the Euclidean distances of the voxels but also on the differences in their activities. The filtering operation is:

$$Q_i = \frac{\sum_{j\in\Omega_i} P_j \cdot g_\sigma(\|i-j\|) \cdot g_\sigma(P_i - P_j)}{\sum_{j\in\Omega_i} g_\sigma(\|i-j\|) \cdot g_\sigma(P_i - P_j)}, \qquad (1)$$

where $\Omega_i$ is the window centered at voxel $i$ whose activity is currently being calculated, $g_\sigma$ is the Gaussian function with standard deviation $\sigma$, and $\|\cdot\|$ denotes Euclidean distance.

### C. Guided Filter

The guided filter [7] is based on a local linear model between a guidance image $G$ and the output image $Q$:

$$Q_i = a_k G_i + b_k, \quad \forall i \in \Omega_k, \qquad (2)$$

where $a_k$ and $b_k$ are coefficients assumed to be constant in the window $\Omega_k$ centered at voxel $k$. Using this model, the output image will be closest to the input if

$$a_k = \frac{\frac{1}{|\Omega|}\sum_{i\in\Omega_k} G_i P_i - \mu_k \overline{P}_k}{\sigma_k^2 + \epsilon}, \qquad (3)$$

$$b_k = \overline{P}_k - a_k \mu_k, \qquad (4)$$

where $\epsilon$ is a regularization constant, $\mu_k$ is the mean and $\sigma_k^2$ is the variance of $G$ in $\Omega_k$, $|\Omega|$ is the number of voxels in $\Omega_k$ and $\overline{P}_k$ is the mean of $P$ in $\Omega_k$.

This way, the output can be calculated as follows:

$$Q_i = \frac{1}{|\Omega|} \sum_{k|i\in\Omega_k} (a_k G_i + b_k) = \overline{a}_i G_i + \overline{b}_i, \tag{5}$$

where $\overline{a}_i = \frac{1}{|\Omega|}\sum_{k\in\Omega_i} a_k$ and $\overline{b}_i = \frac{1}{|\Omega|}\sum_{k\in\Omega_i} b_k$ are the average coefficients of all local windows covering voxel $i$.

The most important task is to choose a guidance so that the outlines of the tissues are kept as sharp as possible while noise is suppressed properly.

One option is to simply use the input image as the guidance. However, because of the local linear relationship between the guidance and the output, an extensively noisy guidance might transfer the noise into the filtered image. If the input is expected to have high noise, denoising might be advantageous before using it as a guidance. In our previous work [9], we applied a Gaussian filter followed by a high-boost filter.

With the spreading of combined PET/CT and PET/MRI scanners, it also becomes possible to use an anatomical image as guidance, thus incorporating tissue boundary information into the filtering. However, since different modalities measure different physical quantities, using their output directly as a guidance can introduce features that are only present in the other modality. To avoid this, we have proposed a joint bilateral filtering algorithm to create a new guidance that uses the anatomical image only indirectly [9]:

$$G_i = \frac{\sum_{j\in\Omega_i} P_i \cdot g(\|i-j\|) \cdot g(A_i - A_j)}{\sum_{j\in\Omega_i} g(\|i-j\|) \cdot g(A_i - A_j)}, \tag{6}$$

where $P$ is the input image, $A$ is the anatomical image, $\Omega_i$ is the window centered at voxel $i$, $g$ is the Gaussian function, and $\|\cdot\|$ denotes the Euclidean distance.

Different guidances can also be combined to form a multi-channel guidance. In the static reconstructions, we worked with a dual-channel guidance whose first channel was the denoised input and the second was the anatomical guidance previously presented. In the dynamic reconstructions, we grouped time frames into three consecutive groups and created a triple-channel guidance by summing the activity images of each of the three frame groups.

### D. Median Filter

The median filter replaces the intensity of each voxel with the median of the neighboring voxels. The method is most effective against salt-and-pepper noise, but it can also eliminate low to moderate levels of Gaussian noise.

### E. Anisotropic Diffusion Filter

The anisotropic diffusion (AD) filter proposed by Perona and Malik [10] describes an iterative diffusion process

$$P_i^{t+1} = P_i^t + \frac{\partial P_i}{\partial t} = P_i^t + \nabla \cdot (g(|\nabla P|)\nabla P), \tag{7}$$

where $t$ is the time or iteration number, $\nabla\cdot$ is the divergence operator, $|\nabla P|$ is the gradient magnitude, and $g$ is the diffusivity, a non-negative monotonically decreasing function with $g(0) = 1$. In our measurements, we have used diffusivity

$$g(|\nabla P|) = \frac{1}{1 + \frac{1}{2}\left(\frac{|\nabla P|}{K}\right)^3}, \tag{8}$$

where $K$ is a threshold that distinguishes noise from the true signal. We estimate $K$ as a weighted average of gradient magnitudes in a local neighborhood of size $10\times10\times10$ voxels. Weights are determined by the similarity of the tissue types of the voxels, that is, the difference between their values sampled from an anatomical (CT, MRI) image. This local estimation is then scaled down by a user-defined detail preservation factor (DPF). Increasing the detail preservation factor decreases threshold $K$ above which features are considered as true signal, therefore preserving finer details.

### F. Non-Local Means Filter

The non-local means (NLM) filter smooths voxel activities by computing a weighted average of activities in a large search window, with weights determined by the similarity of activities in a smaller local neighborhood (also called patch) of the two voxels being compared:

$$Q_i = \frac{1}{\sum_{j\in\Omega_i} w(i,j)} \sum_{j\in\Omega_i} w(i,j) P_i. \tag{9}$$

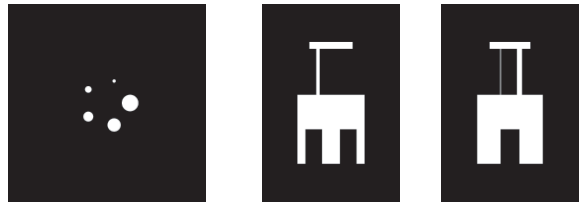
In this equation $\Omega_i$ is the search window centered at voxel $i$ whose activity is currently being calculated and $w(i,j)$ weight is a measure of similarity between the local neighborhoods (patches) of voxels $i$ and $j$ (denoted by $\Psi_i$ and $\Psi_j$):

$$w(i,j) = \exp\left(\frac{-\|P(\Psi_i) - P(\Psi_j)\|_{2,\sigma}^2}{h^2}\right), \tag{10}$$

where $h$ is a user-defined smoothing parameter and $\|\cdot\|_{2,\sigma}^2$ stands for the Gaussian-weighted Euclidean distance with $\sigma > 0$ standard deviation of the Gaussian kernel.

## III. Noise Reduction in Static Reconstructions

Quantitative comparison of the filters was performed on the NEMA NU 4-2008 preclinical phantom [15] (Fig. 3).



(a) Axial (b) Coronal (c) Sagittal

Fig. 3: Slices of the NEMA NU 4-2008 phantom

### A. Image quality

We examined the quality of the images produced by the different filtering algorithms based on two metrics: the *recovery coefficient (RC)* and the *contrast-to-noise ratio (CNR)*.

*1) Recovery Coefficients:* In terms of image quality, the most important parts of the NEMA phantom are the five rods of different diameters, particularly the thinnest rod, which is barely visible. Because of its thinness, there is a risk that this rod will disappear as a result of filtering. Therefore, image quality can be characterized by how well the thinnest rod is reconstructed. This is measured by the recovery coefficient (RC), which is the quotient of the reconstructed and the true activity concentration of the rod. When increasing the blur strength of a filter, the RC should not decrease. We describe blur strength by the percentage standard deviation of the activity in the central uniform region (the large contiguous part in the middle, which is well observable in Fig. 3b and Fig. 3c), i.e. the standard deviation divided by the average activity and then multiplied by 100. The RC values of the examined filters as a function of blur strength are plotted in Fig. 4. To achieve different blur strength, filtering parameters (e.g. regularization, Gaussian standard deviation) were changed incrementally.

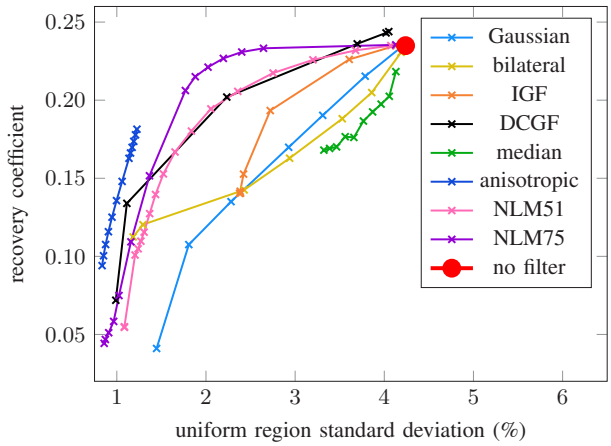



Fig. 4: Recovery coefficient of the thinnest rod as a function of the percentage standard deviation of the uniform region activity. Abbreviations: IGF: input guided Filter, DCGF: dual-channel guided Filter, NLM51: non-local means filter with search window radius of 5 voxels and patch radius of 1 voxel, NLM75: non-local means filter with search window radius of 7 voxels and patch radius of 5 voxels.

It can be seen that the median filter yielded the worst RC values, but the Gaussian filter also significantly reduced the visibility of the rod. When the blurring was slight, the bilateral filter produced worse RC values than the Gaussian filter, but as the blur strength was increased, the RC decreased to a lesser degree with bilateral filtering.

Compared to these methods, guided filtering resulted in extremely good visibility. Even when simply the input image was used as a guidance, the RC was significantly higher than that of the three previously discussed filters, as it can be clearly seen in in Fig. 4. And with the dual-channel guidance, which incorporates anatomical information on tissue boundaries, an even better RC was achieved. When the blurring was slight, not only did the visibility of the rod not decrease, but it in fact increased: an RC value of 0.2439 was attained, whereas without filters, the RC was only 0.2349.

Anisotropic diffusion filtering resulted in strong blurring at all parameter settings. However, in this range of blur strength, this algorithm gave the highest RC values of all filters.

The non-local means filter was examined in two settings that differed in the size of the search window and the patch window. The smaller version (search window radius of 5 voxels and patch radius of 1 voxel) achieved approximately as good visibility as the dual-channel guided filter, while the bigger version (search window radius of 7 voxels and patch radius of 5 voxels) even outperformed it in the mid-range of blur strength. The highest RC produced by NLM filtering was 0.2354, which is slightly higher than without filtering (0.2349), but still lower than the peak RC value of the dual-channel guided filter (0.2439).

When slight blurring was considered, the dual-channel guided filter, in case of moderate blurring, the non-local means filter, and regarding strong blurring, the anisotropic diffusion filter achieved the best recovery coefficients.

*2) Contrast-To-Noise Ratio:* Another important metric for describing image quality is the contrast-to-noise ratio (CNR), which is calculated as [16]

$$CNR = \frac{\mu_{phantom} - \mu_{background}}{\sigma_{background}}, \tag{11}$$

where $\mu_{phantom}$ is the mean activity in the homogeneous phantom, whereas $\mu_{background}$ and $\sigma_{background}$ are the mean and the standard deviation of the activity in the background.

Fig. 5 shows the CNR values of the examined filters as a function of blur strength, i.e. the percentage standard deviation of the activity in the central uniform region.

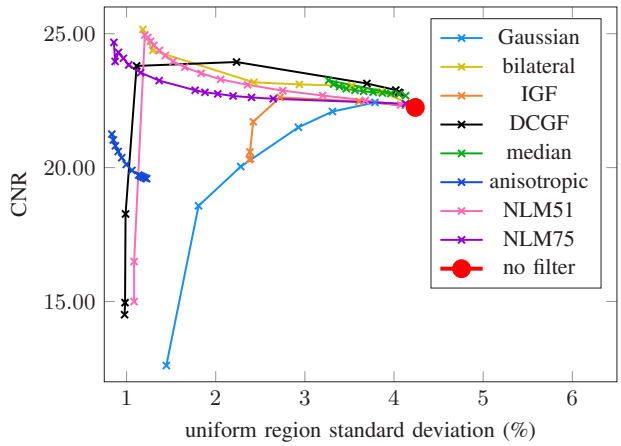



Fig. 5: Contrast-to-noise ratio as a function of the percentage standard deviation of the uniform region activity. Abbreviations: IGF: input guided Filter, DCGF: dual-channel guided Filter, NLM51: non-local means filter with search window radius of 5 voxels and patch radius of 1 voxel, NLM75: non-local means filter with search window radius of 7 voxels and patch radius of 5 voxels.

The Gaussian and the anisotropic filters resulted in very low CNR. However, the bilateral and the median filters, which performed badly regarding the recovery of the thinnest rod, achieved very good contrast-to-noise ratio. The highest

CNR was obtained by the bilateral filter with strong blurring, closely followed by the smaller-window-version of the non-local means filter. The larger-windowed non-local means filter took the lead only when very strong blurring was examined. The dual-channel guided filter also achieved a very good contrast-to-noise ratio – in fact, when the blurring was slight or moderate, it proved to be the best of all filters.

### *B. Runtime*

Runtimes of the Gaussian, the bilateral, the guided, the median, and the non-local mean filters as a function of the filter window radius are summarized in Table I, whereas Table II displays the runtime of the anisotropic filter as a function of the diffusion iteration number.

Gaussian filtering requires negligible time due to the separability of the operation. Bilateral and guided filtering take more time, especially the dual-channel guided filter due to the matrix operations and the preparation time of the anatomical guidance, which involves handling the different resolution of the modalities and performing joint bilateral filtering. However, the execution time of the guided filter is independent of the filter window size. This can make it faster than bilateral filtering for large filter windows.

The increasing runtime of the median filtering was due to the sorting required to determine the median, which was carried out by GPU-based bubble sorting. This is good for small kernels, but for larger ones, other sorting approach should be used instead.

The non-local means filter is very sensitive to both the search window and the path size. If either of the two is large, the runtime can become unacceptably high. Only if the window sizes are small (1-3 voxels) will the runtime be of the same order of magnitude as with bilateral and guided filtering.

Regarding the anisotropic diffusion filter, the execution time is high even if only a few diffusion iterations are carried out. However, it increases linearly with the number of iterations, therefore it can still be faster than the median and the non-local means filters when strong blurring is the goal.

## IV. NOISE REDUCTION IN DYNAMIC RECONSTRUCTIONS

In dynamic reconstructions, filtering is applied to the reconstructed (static) images of each time frame in the last iteration.

Quantitative evaluation of the filters was performed on a rat phantom consisting of four homogeneous regions: body, lung, striatum and cerebellum (Fig. 6).

(a) Axial (b) Coronal (c) Sagittal

Fig. 6: Slices of the ground truth rat phantom

Fig. 7 shows the reconstruction outputs after performing filtering using the different filter algorithms. Filter parameters were fitted to the measurement data, that is, the best achieved results are presented for each filter. The time-activity functions associatied with the images are shown in Fig. 8–Fig. 15. In these graphs, reference activity is indicated by a dashed line, average reconstructed activity by a solid line, and standard deviation of the reconstructed activity by a colored bar around the average (solid) line, where a wider bar means larger standard deviation. Table III presents the quantitative comparison of the filters by displaying the mean and the standard deviation of the reconstructed activity in the striatum (i.e. in the two eye-like region in the output images). The striatum was chosen as the target for the comparison because this is the region where the largest difference can be observed in the filter outputs.

Based on both the output image and the time-activity function, best results were achieved by the non-local means filter with a moderately sized search window and patch window (5 and 1 voxel radius, respectively). It suppressed noise extremely well, as evidenced by the low standard deviation of the reconstructed activity. Boundary edges also remained sharp in all regions. The only place where considerable noise remained is at the edges of the measured volume, which was heavily noisy in the input image due to less data from LORs. This type of noise could only be removed by the median and the anisotropic filters. It should be noted that with a larger search window and patch window, the NLM filter was able to suppress noise better at the volume edges, however, it slightly blurred tissue boundaries.

The second best results were produced by bilateral filtering. Although a few outlier activities can be observed in the output image, the standard deviation of the reconstructed activity is generally low, indicating that most of the noise was suppressed. Furthermore, bilateral filtering preserved tissue boundaries sharp.

Median filtering was not able to completely eliminate the noise, only the outstanding, spike-like values. The less prominent noise values appear as pale dots in the reconstructed image. However, most of the noise at the volume edges was successfully removed. On the other hand, median filtering undesirably reduced the size of the small, but highly active striatum region by replacing voxel activities at the edge of the striatum with the surrounding lower activity of the body.

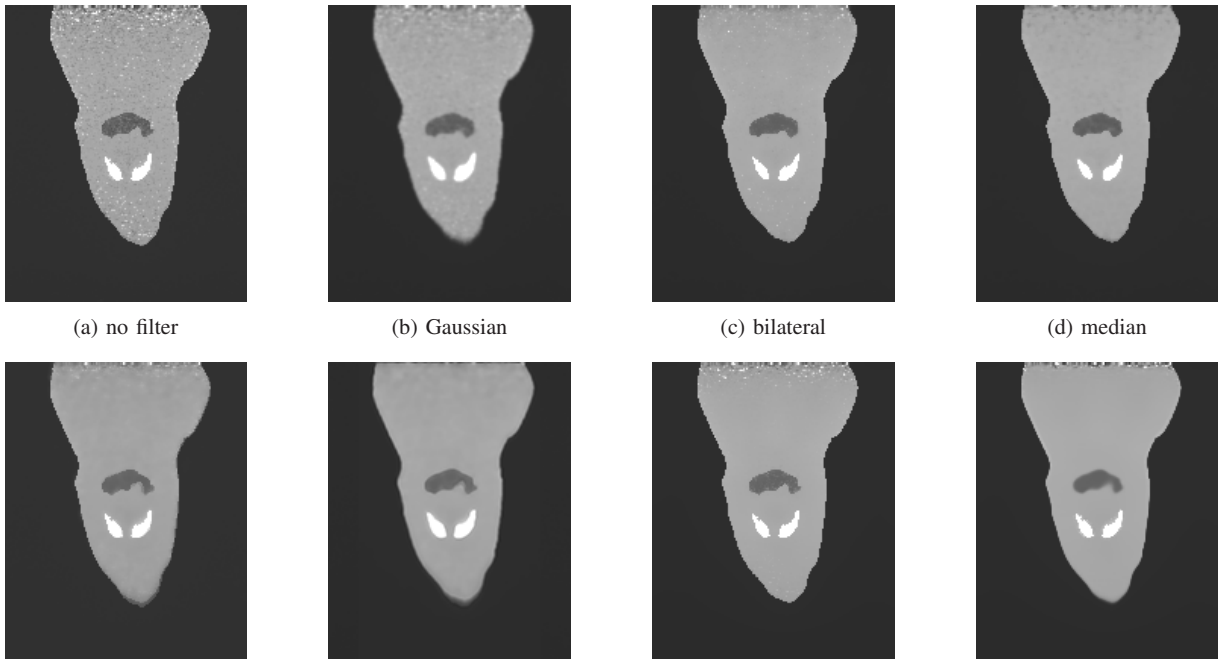
For guided filtering, we grouped time frames into three consecutive groups and created a triple-channel guidance by summing the activity images of each of the three frame groups. Using this guidance, guided filtering managed to eliminate noise in the body and the cerebellum, but achieved less good results in the lung and the striatum. In addition, it slightly blurred tissue boundaries in some places, such as the right part of the striatum and the nose of the rat.

The anisotropic diffusion filter reduced noise very effectively, obtaining an almost homogeneous activity even at volume edges. However, region boundaries are not as sharp as with other filters, a slight effect can be observed as if tissues had double edges.

Finally, the Gaussian filter was unable to eliminate the noise, only to blur it, compromising region boundaries too.

| $r$ | Gaussian | bilateral | IGF | DCGF | median | NLM1 | NLM3 | NLM5 |
|---|---|---|---|---|---|---|---|---|
| 1 | 0.0089 | 0.3535 | 0.4857 | 1.2466 | 0.0410 | 0.0541 | 0.6118 | 2.4043 |
| 2 | 0.0090 | 0.3533 | 0.4865 | 1.2488 | 1.0632 | 0.2212 | 2.9073 | 11.1695 |
| 3 | 0.0092 | 0.3542 | 0.4867 | 1.2464 | 9.2433 | 0.5928 | 7.9972 | 30.8299 |
| 4 | 0.0094 | 0.3596 | 0.4880 | 1.2462 | 44.4551 | 1.3093 | 16.9861 | 65.6742 |
| 5 | 0.0096 | 0.3754 | 0.4867 | 1.2468 | 178.0231 | 2.4917 | 31.0007 | 120.1560 |
| 6 | 0.0098 | 0.4003 | 0.4866 | 1.2465 | 478.2444 | 4.1584 | 51.1137 | 192.7670 |
| 7 | 0.0101 | 0.4322 | 0.4861 | 1.2462 | 1116.0873 | 6.3966 | 78.8206 | 301.5630 |
| 8 | 0.0103 | 0.4767 | 0.4859 | 1.2473 | 2308.0518 | 9.3278 | 115.0860 | 440.2210 |
| 9 | 0.0105 | 0.5293 | 0.4860 | 1.2517 | 4412.3638 | 13.0100 | 157.4710 | 616.5830 |
| 10 | 0.0107 | 0.5968 | 0.4868 | 1.2454 | 7911.3560 | 17.5618 | 217.8390 | 833.3860 |

TABLE I: Execution time of the filtering algorithms in seconds at different filter window radiuses, averaged over 100 runs. Measurements were run on a NVIDIA TITAN V graphics card [17]. For the dual-channel guided filter, the preparation of the guidance is also included at the indicated time. For non-local means filter, the patch window radius was fixed and the search window radius was changed. Abbreviations: IGF: input guided filter, DCGF: dual-channel guided filter, NLM1, NLM3, NLM5: non-local means filter with patch radius of 1, 3, and 5 voxels, respectively.



(a) no filter (b) Gaussian (c) bilateral (d) median

(e) guided (f) anisotropic (g) NLM, search window radius $= 5$, patch radius $= 1$ voxels (h) NLM, search window radius $= 7$, patch radius $= 5$ voxels

Fig. 7: Post-reconstruction filtering of a dynamic rat phantom using different filter algorithms

| iterations | anisotropic |
|---|---|
| 20 | 3.9669 |
| 40 | 5.4686 |
| 60 | 6.9714 |
| 80 | 8.4806 |
| 100 | 9.9869 |
| 120 | 11.4830 |
| 140 | 12.9974 |
| 160 | 14.4686 |
| 180 | 15.9919 |
| 200 | 17.5020 |

TABLE II: Execution time of the anisotropic diffusion filtering in seconds as a function of the iteration number of the diffusion process, averaged over 100 runs.

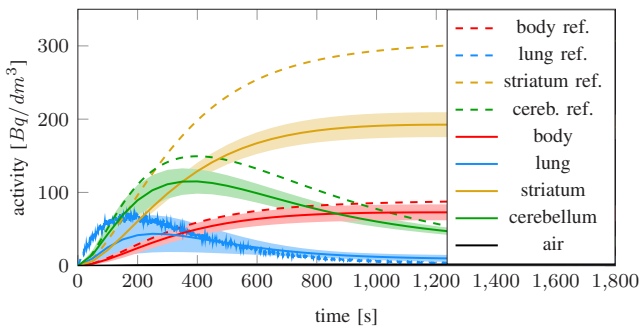



Fig. 8: Time-activity function without filters

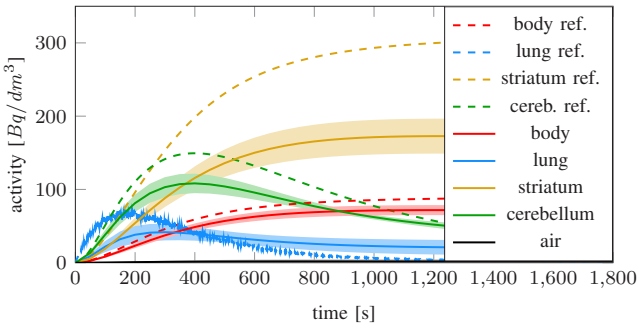



Fig. 9: Time-activity function after Gaussian filtering with $\sigma = 0,75$ standard deviation

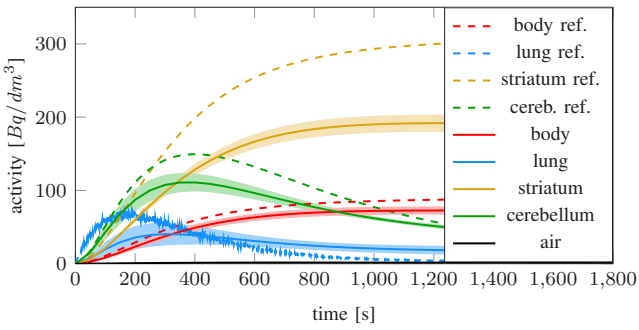



Fig. 10: Time-activity function after bilateral filtering with $\sigma = 2$ spatial standard deviation

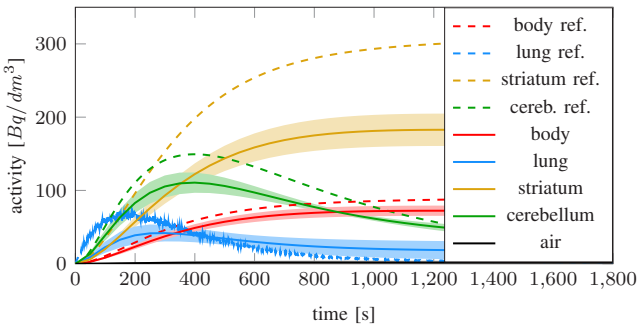



Fig. 11: Time-activity function after guided filtering with $\epsilon = 1000$ regularization

| | mean | stddev |
|---|---|---|
| ground truth | 242.7996 | 0.0000 |
| no filter | 155.9013 | 13.8608 |
| Gaussian | 139.8609 | 19.2666 |
| bilateral | 155.2769 | 9.5501 |
| median | 142.8997 | 27.1341 |
| guided | 147.9435 | 17.8524 |
| anisotropic | 133.6422 | 19.6897 |
| nlm51 | 155.9950 | 6.8938 |
| nlm75 | 155.0453 | 13.7324 |

TABLE III: Mean and standard deviation of the activity in the striatum using different filter algorithms, averaged over the entire measurement time

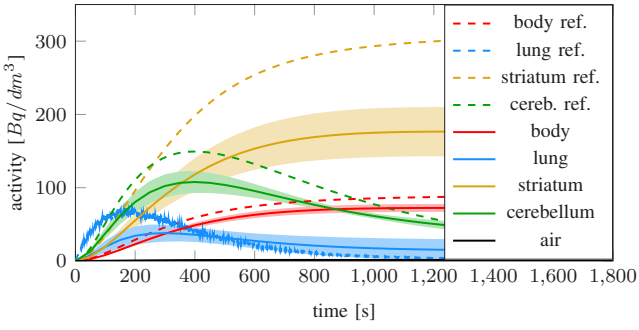



Fig. 12: Time-activity function after median filtering

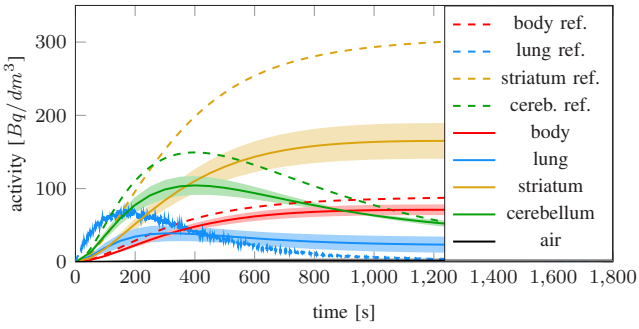



Fig. 13: Time-activity function after anisotropic filtering with detail preservation factor = 2

## V. CONCLUSION

This paper examined post-reconstruction filtering for PET image denoising, comparing the most commonly used filters in terms of image quality and runtime. Quantitative analysis was performed in both a static and a dynamic reconstruction.

In the static reconstruction, the best image quality was achieved by the dual-channel guided filter, followed closely by the non-local means filter. However, both of these filters have a relatively high runtime. When time is important, a single-channel guided filter (e.g. when the guidance is the input image) should be considered.

In the dynamic reconstruction, the best results were obtained by the non-local means filter with a moderately sized search window and patch window. Having slightly more noise, but preserving edges just as sharp, the bilateral filter achieved the second best results. This is also advantageous because bilateral filtering is relatively fast (Table I).

It can be concluded that the best filtering method depends on the measurement data and the reconstruction settings. However, considering both our static and dynamic reconstruction studies, the non-local means filter proved to be the most promising method. In both scenarios, it suppressed noise extremely well and kept tissue boundaries sharp. Its weak point is the high runtime, which should be avoided by setting the search window and the patch window small. In the dynamic reconstruction study, decreasing the window sizes even resulted in a more accurate output image.

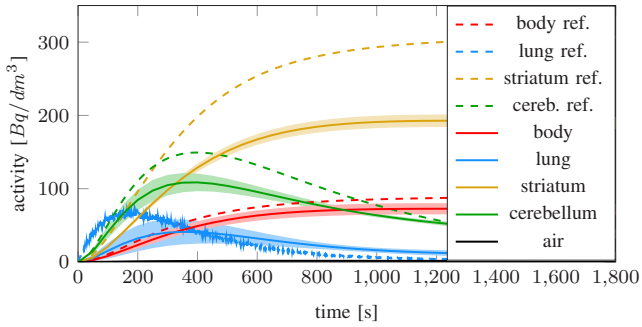



Fig. 14: Time-activity function after non-local means filtering with search window radius $= 5$ voxels, patch radius $= 1$ voxel, and $h = 5000$ smoothing

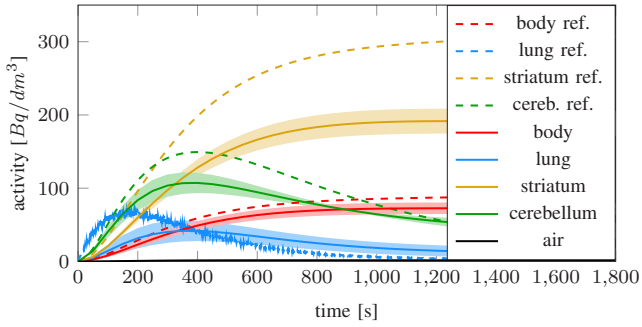



Fig. 15: Time-activity function after non-local means filtering with search window radius $= 7$ voxels, patch radius $= 5$ voxels, and $h = 2500$ smoothing

## ACKNOWLEDGMENT

The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications), Infocommunications), by the ÚNKP-19-1 New National Excellence Program of the Ministry for Innovation and Technology, and by the OTKA K–124124.

## REFERENCES

[1] J. Nuyts and J. A. Fessler, "A penalized-likelihood image reconstruction method for emission tomography, compared to postsmoothed maximumlikelihood with matched spatial resolution," *IEEE Transactions on Medical Imaging*, vol. 22, no. 9, pp. 1042–1052, Sep. 2003, DOI: 10.1109/TMI.2003.816960.

[2] D. Yu and J. Fessler, "Edge-Preserving Tomographic Reconstruction with Nonlocal Regularization," *Medical Imaging, IEEE Transactions on*, vol. 21, pp. 159–173, Mar. 2002, DOI: 10.1109/42.993134.

[3] T. J. Herbert, "Statistical stopping criteria for iterative maximum likelihood reconstruction of emission images," *Physics in Medicine and Biology*, vol. 35, no. 9, pp. 1221–1232, Sep. 1990, DOI: 10.1088/0031-9155/35/9/003.

[4] W. K. Pratt, Digital image processing: PIKS Scientific inside, 4th ed. Wiley-Interscience, 2007.

[5] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*. Narosa Publishing House, Jan 1998, pp. 839–846, DOI: 10.1109/iccv.1998.710815.

[6] L. Papp, G. Jakab, B. T´oth, and L. Szirmay-Kalos, "Adaptive bilateral filtering for PET," in *IEEE Nuclear science symposium and medical imaging conference*, ser. MIC'14, 2014, pp. M18–104.

[7] K. He, J. Sun, and X. Tang, "Guided Image Filtering," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, pp. 1397–1409, Jun. 2013, DOI: 10.1109/TPAMI.2012.213.

[8] J. Yan, J. C.-S. Lim, and D. W. Townsend, "MRI-guided brain PET image filtering and partial volume correction," *Physics in Medicine and Biology*, vol. 60, no. 3, pp. 961–976, Jan. 2015, DOI: 10.1088/0031-9155/60/3/961.

[9] D. Varnyú and L. Szirmay-Kalos, "Guided Filtering and Partial Volume Correction for Positron Emission Tomography," in *Proceedings of the Workshop on the Advances in Information Technology, ser. Workshop on the Advances in Information Technology (WAIT)*, B. Kiss and L. Szirmay-Kalos, Eds., no. ISBN 978-963-421-802-9, Budapest, Hungary, Jan. 2020, pp. 89–98.

[10] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 629–639, 1990, DOI: 10.1109/34.56205.

[11] L. Szirmay-Kalos, M. Magdics, and B. Tóth, "Volume enhancement with externally controlled anisotropic diffusion," *The Visual Computer*, vol. 33, pp. 331–342, 2017, DOI: 10.1007/s00371-015-1203-y.

[12] W. Qi, T. Xia, X. Niu, C. Ji, M. Winkler, E. Asma, and W. Wang, "A non-local means post-filter with spatially adaptive filtering strength for whole-body PET," in *2015 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*. IEEE, Oct. 2015, DOI: 10.1109/nssmic.2015.7582060.

[13] A. Buades, B. Coll, and J.-M. Morel, "A Non-Local Algorithm for Image Denoising," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. IEEE, 2005, DOI: 10.1109/cvpr.2005.38.

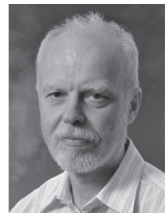
[14] C. Chan, R. Fulton, D. D. Feng, and S. Meikle, "Median non-local means filtering for low SNR image denoising: Application to PET with anatomical knowledge," in *IEEE Nuclear Science Symposuim Medical Imaging Conference*, Oct. 2010, pp. 3613–3618, DOI: 10.1109/NSSMIC.2010.5874485

[15] A. L. Goertzen et al., "NEMA NU 4-2008 comparison of preclinical PET imaging systems," The Journal of Nuclear Medicine, vol. 53, no. 8, pp. 1300–1309, Aug. 2012, DOI: 10.2967/jnumed.111.099382.

[16] J. Yan, J. Schaefferkoetter, M. Conti, and D. Townsend, "A method to assess image quality for Low-dose PET: analysis of SNR, CNR, bias and image noise," *Cancer Imaging*, vol. 16, p. 26, 2016, DOI: 10.1186/s40644-016-0086-0.

[17] "NVIDIA Titan V GPU specifications," http://www.gpuzoo.com/GPU-NVIDIA/Titan_V.html, accessed on 28/10/2019.

**Dóra Varnyú** is an MSc student at Budapest University of Technology and Economics. She is a member of the Balatonfüred Student Research Group. She has contributed to numerous international conferences in the field of medical imaging and computer graphics. Her research focuses on positron emission tomography and has so far covered filtering techniques and motion compensation.



**László Szirmay-Kalos** was graduated from Budapest University of Technology and Economics in 1987, received Ph.D. and Doctor of Science degree from the Hungarian Academy of Science in 1991 and in 2001, respectively. He is currently a full professor of computer graphics at Budapest University of Technology. His research interests include rendering, Monte Carlo methods and medical imaging. He is the fellow of Eurographics. His web page is https://www.iit.bme.hu/users/dr-szirmay-kalos-

IFIP/IEEE International Symposium on Integrated Network and Service Management (IM 2021)
17-21 May 2021 // Bordeaux, France
http://im2021.ieee-im.org

## Call for Papers

The 17th IFIP/IEEE Symposium on Integrated Network and Service Management (IM 2021) will be held on May 17-21, 2021, in Bordeaux, France. Held in odd-numbered years since 1989, IM 2021 follows the 33 years tradition of NOMS and IM as the primary IEEE Communications Society's forum for technical exchange on management of information and communication technology focusing on research, development, integration, standards, service provisioning, and user communities.

IM 2021 will focus on the theme of "**Intelligent Management of Open and Highly Programmable Networks**". It aims to capture recent results, emerging approaches and technical solutions for dealing with resilience and sustainability of network and service management in highly dynamic environments. Furthermore, the sudden and important uptake of work-from-home and the promise of the post-digital era are rising new challenges and exacerbating the dynamic nature of the communication infrastructure, pushing additional requirements on networks in terms of quality, resilience and sustainability. This is also accelerating the adoption of Artificial Intelligence advances along with the openness and programmability of network infrastructures, to make these more flexible and adaptive. In addition to this theme, IM 2021 will cover a wide range of topics of interest within the field in integrated network and service management. IM 2021 will offer five types of sessions: technical, experience, poster, panel and dissertation. High quality will be assured through a well-qualified Technical Program Committee and stringent peer review of paper submissions.

### Topics of Interest

Authors are invited to submit papers that fall into or are related to the topic areas that are listed below. In addition, we invite submissions of proposals for demonstrations, exhibits, technical panels, tutorials and workshops, as well as experience session papers and dissertation papers.

**Management and Control of Networks**
- Enterprise and Campus Networks
- Data Center Networks
- Industrial Networks and TSN
- Cyber Physical Systems
- IP Networks
- 5G/6G,
- Wireless & Mobile Networks, including optical, overlay, access, SCADA, IoT, VANET, etc.

**Programmable Networks and Automation**
- Software-Defined Networking
- Network Virtualization
- Network (Data, Control, Management planes) Programmability
- Network Slicing
- Management & Orchestration
- Network Functions Virtualization/Service Function Chaining
- Cloud Native Networking, DevOps, Network Automation, Zero-Configuration Networking
- Network Telemetry

**Management and Control of Communication Services**
- Information Technology Services
- Virtual Networking Services
- XaaS and Cloud Services
- Multimedia Services
- Content Delivery Services
- Social Networking Services
- Security Services
- IoT Services
- Network Resilience and High-Precision Networks

**Management Algorithms and Architectures**
- Centralized Management
- Distributed Management
- Autonomic Networks and Self-Management
- Management Protocols
- Fog and Mobile Edge Computing
- Lambda Functions and Elastic Management
- Data Analytics for Management
- Policy-Based Management
- Intent-Based Management
- Network Optimization

**Management Functions and Practical Approaches**
- FCAPS: Fault, Configuration, Accounting, Performance and Security
- Case Studies and Practical Experiences
- Business-Driven Management Services
- Measurement and Validation
- SLA Management, QoS and QoE
- Energy Management
- Deployment of Services
- Operations Support Systems

**Artificial Intelligence Techniques for Network and Service Management**
- Management with AI
- Markov Chains and Management
- Machine Learning and Deep Learning
- Management with Big Data
- Mobile Agents

**Management Efforts for Pandemics and Crisis Situations (COVID-19)**
- Contact and Activity Tracing
- Network/Service Management Support
- Network Measurements
- Network Adaptation
- Case Studies and Practical Experiences

Contact TPC Co-chairs for more information: im2021tpc@gmail.com
Submission link: https://submissoes.sbc.org.br/home.cgi?c=3614

### Important Dates

Paper Submission Deadline: September 20, 2020
Notification of Acceptance: December 10, 2020
Camera-Ready Submission: January 29, 2021

### General Co-chairs

Toufik Ahmed, Bordeaux INP, France
Olivier Festor, University of Lorraine, France

### TPC Co-chairs

Yacine Ghamri-Doudane, La Rochelle University, France
Alberto E. Schaeffer-Filho, UFRGS, Brazil
Joon-Myung Kang, Google, USA

# 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 EasyChair system. The link for submission can be found on the journal's website:

www.infocommunications.hu/for-our-authors

If you have any question about the journal or the submission process, please do not hesitate to contact us via e-mail:

Pál Varga – Editor-in-Chief:
pvarga@tmit.bme.hu

Rolland Vida – Associate Editor-in-Chief:
vida@tmit.bme.hu

IEEE ICC®

**IEEE International Conference on Communications (ICC 2021)**
14-18 June 2021 // Montreal, QC, Canada
*Connectivity – Security – Privacy*

IEEE | IEEE ComSoc IEEE Communications Society

# Call for Papers for
# *Symposium on Selected Areas in Communication*
# *Track on Machine Learning for Communications Track*

## Track Co-Chairs


- **Marios Kountouris, EURECOM, France, marios.kountouris@eurecom.fr**
- **Fernando Perez-Cruz, Swiss Data Science Center, Switzerland, fernando.perezcruz@sdsc.ethz.ch**


## Topics of Interest

We invite submissions of unpublished works on the theory and application of ML to communications, as well as the proposals of new ML algorithms or architectures. We do not restrict the type of ML techniques. A non-exhaustive list of relevant topics is given below.

- ML empowered transceiver design and channel coding
- ML driven techniques for radio environment awareness and spectrum access
- ML based enhancements for channel modeling, including non-traditional communications mediums (optical, quantum, molecular, biological, etc.)
- ML techniques for nonlinear signal processing
- Distributed/decentralized machine learning, decision making, and edge intelligence
- ML framework for joint communication, computation, and control
- (Deep) Reinforcement Learning for resource management & optimization
- Low-complexity/low-power deep learning hardware implementations
- Transfer learning, few-shot learning, and meta-learning in communication systems
- Privacy and security preserving training over communications networks

## Important Dates

**Paper Submission:** 12 October 2020
**Notification:** 25 January 2021
**Camera Ready and Registration:** 22 February 2021

## How to Submit a Paper

All papers for technical symposia should be submitted via EDAS. Full instructions on how to submit papers are provided on the IEEE ICC2021 website: https://icc2021.ieee-icc.org/

# 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: **GÁBOR MAGYAR, PhD** • *elnok@hte.hu*

Secretary-General: **ERZSÉBET BÁNKUTI** • *bankutie@ahrt.hu*

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*