Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

September 2021 Volume XIII Number 3 ISSN 2061-2079

MESSAGE FROM THE EDITOR-IN-CHIEF

PAPERS FROM OPEN CALL

CALL FOR PAPERS / PARTICIPATION

ADDITIONAL

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, 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

From traffic analysis to system security: broad interest within the Infocommunications domain

Pal Varga

THE TERM ”traffic analysis” may be misleading in this issue of Infocommunications Journal. Civilians naturally think it is something about analyzing vehicular transport on the roads – although for ICT practitioners it always has been about telco- or computer network traffic. Surprisingly though, the first two articles in this issue are actually discussing road transport traffic analysis. The methods we use in the infocommunications domain is now applied to the transport domain by our very own experts in the communications society. The third article in this issue is indeed, on (cellular) mobile network traffic. Mobile network in the the sense that the user equipment can be mobile; yet another way to get confused with transportation systems. The current issue of the journal features eight papers and 92 pages if counting the front and back covers as well. This makes the current issue the thickest so far – but not only in volume. Let us have a brief overview of the papers in this issue.

Attila Nagy and his co-authors aim to detect incidents on the roads that lead to congestion. Their new, Transient-based Automatic Incident Detection (TBAID) method uses a novel approach to detect the occurrence of incidents, using new features such as speed, flow and occupancy. Their results showed that this method performed better than the currently available ones in terms of both speed and reliability on traffic data collected from freeways. They also made the data-set available for further, open analysis and comparisons.

In their paper, Mehran Amini et. al. describe a new, macroscopic model based on fuzzy cognitive map for road traffic flow simulation. They applied fuzzy cognitive maps (FCM) reasoning on historical data collected from the e-toll dataset of Hungarian networks of freeways. Through the customized scenarios, macroscopic modeling objectives such as predicting future road traffic flow state, route guidance, freeway geometric characteristics indication, and effectual mobility can be evaluated by using their method.

Regarding the dynamic management of 5G network resources, Khalil Mebarkia and Zoltán Zsóka present the QoS impacts of slice traffic limitation. They propose different policies for setting up the parameters of the service function chaining methods. The model behind the methods ignores the load and latency details or limitations of VNFs, but considers link capacities and network loads coming from the different slices, which share the available resources according to the implemented queueing. This allows the systematic evaluation of QoS properties that can be experienced on the links or by the service requests.

Yahieal Alnaiemy and Lajos Nagy descibe their design for a novel UWB monopole antenna structure with reconfigurable band notch characteristics based on PIN diodes. The proposed antenna is comprised of a modified circular patch and a partial ground plane. The band-notch characteristics are achieved by etching a slot on the partial ground plane and inserting three PIN diodes (allowing reconfigurability for eight states with UWB) into the slots for adjusting the operating antenna bands.

The Hungarian research organizations joined forces under the Quantum Technology National Excellence Program to stay in the frontline of the Quantum Key Distribution (QKD) domain. In their paper, Márton Czermann et. al. demonstrate the first successful quantum key distribution over physical layer in accordance with the truth table of BB84 protocol in the country. Part of the deterministic tests they achieved 97.49% as the best individual performance among base pairings.

Gábor Árpád Németh and Máté István Lugosi present a new, heuristic algorithm for the All-Transition-State criteria of deterministic finite state machine specifications. The length of the resulting test suite and its fault coverage can be fine-tuned with the three different versions of their algorithm (standard, iterative with and without an iteration limit) allowing the test engineer to find a suitable trade-off between the overall length of the test suite and fault coverage.

Silia Maskuti et. al. present their new results towards security mitigation in SoS using a generic autonomic management system to assist engineers in developing self-adaptive systems. They propose a generic autonomic management system (GAMS) that automatically tracks runtime uncertainties and adapts System of Systems (SoS) settings without human intervention.

In their paper, Matthias Maurer and his co-authors investigate the possibility to create a predictive maintenance framework using only easily available log data based on a neural network framework for predictive maintenance tasks. They outline the advantages of the ALFA (AutoML for Log File Analysis) approach, which are high efficiency in combination with a low entry border for novices, among others.

Pal Varga received his Ph.D. degree from the Budapest University of Technology and Economics, Hungary. He is currently an Associate Professor at the Budapest University of Technology and Economics and also the Director at AITIA International Inc. His main research interests include communication systems, Cyber-Physical Systems and Industrial Internet of Things, network traffic analysis, end-to-end QoS and SLA issues – for which he is keen to apply hardware acceleration and artificial intelligence, machine learning techniques as well. Besides being a member of HTE, he is a senior member of IEEE, where he is active both in the IEEE ComSoc (Communication Society) and IEEE IES (Industrial Electronics Society) communities. He is Editorial Board member of the Sensors (MDPI) and Electronics (MDPI) journals, and the Editor-in-Chief of the Infocommunications Journal.

Transient-based automatic incident detection method for intelligent transport systems

Attila M. Nagy, Bernát Wiandt and Vilmos Simon

***Abstract*—One of the major problems of traffic in big cities today is the occurrence of congestion phenomena on the road network, which has several serious effects not only on the lives of drivers, but also on city inhabitants. In order to deal with these phenomena, it is essential to have an in-depth understanding of the processes that lead to the occurrence of congestion and its spilling over into contiguous areas of the city.**

One of the main causes of congestion phenomena is unexpected traffic incidents on major roads and urban freeways, the rapid and reliable detection of which can help reduce negative impacts. Researching Automatic Incident Detection (AID) has a long history that has again become one of the main subjects of research with the rise of new machine learning methods.

Our article presents a new Transient-based Automatic Incident Detection (TBAID) method we have developed, which uses an approach not yet seen in professional literature to detect the occurrence of incidents. The results of our detailed analysis showed that our method performed better than the methods currently available in terms of both speed and reliability on traffic data collected from freeways.

We also created a new dataset for the examination of our method, because the datasets used in previous research were either too small or not publicly available. Our dataset contains 452 incidents and data measured with dual loop traffic detectors from the immediate vicinity of incidents, which, to the best of our knowledge, is the largest publicly available incident dataset to date.

***Index Terms*—automatic incident detection, time series analysis, congestion, smart cities**

I. Introduction

One of the major problems related to transportation in major cities around the world is the phenomenon of traffic jams and congestion occurring on major roads and urban freeways. Congestion has a serious impact not only on the lives of vehicle drivers, but also on the lives of every inhabitant of the city. Congestion increases energy and fuel consumption, as well as harmful emissions [1], [2]. Other research has focused on the physiological effects of congestion. Air pollution associated with congestion has been shown to increase the chances of developing allergies [3] and to aggravate the symptoms of people who are sensitive to them. In addition, studies have shown that congestion also increases the risk of heart attacks [4].

The negative effects listed above illustrate the significance of avoiding and possibly eliminating congestion, as they harm the health of citizens in addition to causing significant economic damage. A reduction of congestion would bring serious economic and social benefits [5].

Intelligent city management systems can provide solutions to these problems or at least significantly reduce negative impacts with the help of Intelligent Transportation Systems (ITS) [6]. The task of these systems is to continually monitor the traffic and to provide information to the urban transport infrastructure designers and operators based on the collected data, as well as to manage the automated allocation of resources, for example, opening or closing new lanes, adapting traffic lights to current traffic conditions [7] or assisting route planning applications with accurate forecasts.

There are countless reasons for the occurrence of congestion, only some of which can be predicted. Unpredictable congestion phenomena typically do not repeat and are usually caused by unexpected traffic incidents. Research has shown that traffic incidents account for at least 60% of non-recurring congestion [8], [9].

In order to reduce the negative effects of unexpected traffic incidents, it is essential that intelligent city management systems are able to respond as quickly as possible to unexpected situations. In addition to providing useful information to city traffic management, quick and reliable AID can also provide new data for route planning and traffic forecasting algorithms, along with being an important data source for dynamic traffic light control systems.

AID is a long-established area of research that has come back into focus now that new types of data sources and data analysis methods, as well as increasingly used artificial intelligence-based solutions, have become wide-spread [10]. From the start, it has been a major challenge for researchers to address the contradiction between the accuracy and the speed of detection. Looking at the performance of the methods found in professional literature, it can be concluded that although the methods are capable of high detection rates, even close to 100%, they are also very slow to detect incidents, or they send a number of false alarms. The opposite of this phenomenon can also be observed, with rapid detection being achieved but accompanied by a low accuracy of less than 70%. It is important to point out that frequent false detection makes the task of traffic management extremely difficult. False detection can result in incorrect reallocation of resources and modification of traffic light schedules, which can upset the otherwise normal pace of traffic.

Another major challenge with AID is obtaining a suitable

Attila M. Nagy is with the Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Techonolgy and Economics, Hungary, e-mail: anagy@hit.bme.hu.

Bernát Wiandt is with the Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Techonolgy and Economics, Hungary, e-mail: bwiandt@hit.bme.hu.

Vilmos Simon is with the Department of Networked Systems and Services, Faculty of Electrical Engineering and Informatics, Budapest University of Techonolgy and Economics, Hungary, e-mail: svilmos@hit.bme.hu.

DOI: 10.36244/ICJ.2021.3.1

dataset. Since incidents are rarely occurring events, collection is difficult in large quantities. It is also important that we have information not only about incidents, but also about traffic data in their immediate vicinity. Unfortunately, the datasets used for research in professional literature contain a small number of incidents (10-30), which is not sufficient for the artificial intelligence models used today, or the dataset has not been made publicly available.

In this article, we would like to offer a solution to the two challenges mentioned above. First, we created an incident dataset containing data from 452 incidents as well as traffic detector data from the immediate vicinity of incidents for the investigated time period. The dataset has been made publicly available to make our results reproducible and to assist related scientific research.

Using the completed dataset, we developed a new AID model. To do this we applied a new approach, in which we used state-of-the-art machine learning tools and developed new, complex features that focus on detecting transient phenomena caused by incidents in traffic data. Our detailed analysis showed that the model we developed can surpass the methods from professional literature in accuracy as well as speed, with low false alarm rates.

The remainder of this article contains the following sections. In Section II, we present related works found in professional literature. In addition to presenting previously developed AID methods, we also place considerable focus on describing the key features of incidents. The method we have developed is described in Section III. The evaluation of TBAID is performed in Section IV, where we compare it with the results of several machine learning models and previous AID methods. We end our article with a short conclusion in Section V.

II. Related research

For decades, researchers and city managers have been working on ways to automate the detection of traffic incidents. The reliable and fast Automatic Incident Detection (AID) allows city managers to take preventive action to avoid congestion, as well as route planners and forecasting systems to use this additional information to improve planned routes and forecasts.

The proper implementation of incident detection requires understanding and examination of traffic phenomena caused by incidents. Therefore, in Section II-A we focus on presenting the features defined by professional literature that are currently used for implementing AID. The main AID methods from professional literature are then described in Section II-B.

A. Incidents

An incident is defined as any non-recurring event on a road network that reduces the capacity of a given road segment. An incident can be an accident, a pulled over or broken down vehicle, traffic hazards, debris on the road, fallen cargo, road network maintenance or refurbishment and other special, non-emergency events [11], [9]. Events in the previous list are referred to as incident types.

To categorize incidents, the Traffic Incident Management Handbook (TIMH) [12] defines an incident profiling and classification procedure based on the type, location (has it blocked a lane?) and duration of the incident. The incident classification shows that we have data for 70% of all incidents, of which 80% are related to vehicles pulled over, 10% are accidents and the remaining 10% are classified in other categories. It can be seen that in all cases the incidents that block lanes are causing relatively large delays, but the incidents at the side of the road can also cause measurable capacity reductions. Accidents blocking multiple lanes cause considerably large delays. This means that incidents that have no effect on traffic development cannot be detected from traffic data, so there will definitely be a subset of incidents that are impossible to detect with a traffic-data based AID.

The Manual on Uniform Traffic Control Devices (MUTCD) [13] compiles incident types into three main categories based on a similar set of criteria. *Major incidents* last for at least 2 hours and are typically fatal accidents or other incidents involving dangerous substances that are difficult to clean getting on the road. When this happens, it is often necessary to close all lanes (interestingly, these 2 hours are not in line with the maximum 90-minute value in TIMH [12]). The length of *intermediate incidences* falls between 30 minutes and 2 hours. This may require the complete closing of the given road segment, but partial roadblocks are more common. *Minor incidences* are those of less than 30 minutes that rarely require lane closure. Typically this includes vehicles that are pulled over or small collisions.

Recently, the length of incidents has been the subject of several studies [14], [15], [16], as this may be valuable information for road network management organizations, route planning algorithms or traffic forecasting services. These researches have found that different types of incidents have different lengths that correspond to them. A study conducted in Australia [15] showed that the accidents included in the study lasted on average 43 minutes and the incidents related to pulled-over vehicles lasted on average 41 minutes. Hazards have the longest lasting effects, with an average length of 74 minutes. Another interesting observation was that incidents last longer on weekdays than on weekends.

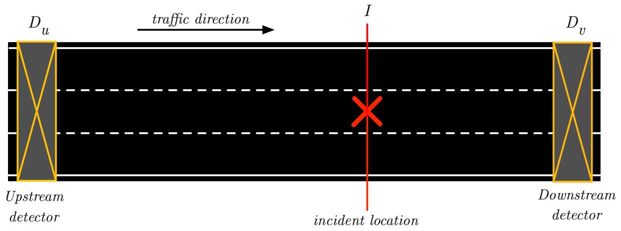


Fig. 1: Configuration method for examining traffic incidents.

A common examination method for traffic incidents is illustrated in Figure 1, where incidents are detected with data from traffic detectors [17], [18], [19], [20], [21]. Of course, there are other methods [22], [23], which determine the occurrence of an incident from travel time data or speed/acceleration data extracted from vehicle trajectories, but we won't deal with these approaches here.

Incident I is detected using two traffic detectors: D_u (*upstream*) and D_d (*downstream*). In relation to traffic direction, the *upstream* detector is located in the pre-incident area, while the *downstream* detector is monitoring traffic in the post-incident segment. Using *upstream* and *downstream* detectors, several articles break down the investigated road network into segments, where detection is carried out.

The detectors typically measure *flow*, *speed* and *occupancy* values, which are always determined for consecutive time intervals of a fixed length. Typical time intervals include 30 seconds, 5 minutes and 1 hour. *Flow* represents the number of vehicles per time unit (veh/h) and *speed* represents the average speed of vehicles passing by a detector for a given time interval. *Occupancy* indicates the percentage of time vehicles were over a detector for a given time interval.

In case of incident I, different phenomena may be observed on the *upstream* and *downstream* detectors [24].

In order to better illustrate the differences in the forming traffic patterns between *upstream* and *downstream* detectors, a comparison of different measured metrics can be found in Figure 2 for three different traffic demands. If, as a result of the incident, the capacity of the affected road segment is reduced at the location of the incident, vehicles will start to pile up when traffic demand rises above this amount on the pre-incident segment towards the *upstream* detector. As soon as the effect reaches the D_u *upstream* detector, significantly reduced flow and *speed* values can be measured (Figure 2c and Figure 2e), while occupancy, in contrast, increases (Figure 2a). However, it is important to note that if the traffic demand is low enough, it is impossible to detect the incident from traffic detector data, since even with reduced capacity, the road segment can serve the current traffic demand.

In the meantime, the measured speed values on the D_d *downstream* detector start to increase up to the free-flow speed (Figure 2d).

Free-flow speed is the speed at which vehicle drivers can go when other vehicles do not impede their movement [25]. The vehicles are congested before the incident, therefore the flow and occupancy values measured on the *downstream* detector will show a declining trend compared to the pre-incident values (Figure 2f and Figure 2b). Another important observation is that the effect on the *upstream* detector appears slower than on the *downstream* detector, as the congestion phenomenon propagating in the *upstream* direction is moving slower than the vehicles leaving the incident.

As a first step we looked at the occupancy time series. Figure 2a shows a dramatic increase in occupancy in case of all three traffic demands after the occurrence of the incident with variable delay. An interesting observation of the *upstream* is that the time it takes for the effect of the incident to appear depends on the traffic demand. The higher the traffic demand, the faster the effect appears. This is logical, since vehicles are congesting faster behind each other.

It is important to note that depending on the position of the incident between two measuring stations, the effect of the incident on the detectors appears with different delays.

In contrast to upstream, the phenomenon that appears downstream (Figure 2b) appears quite quickly, but even though the effect can be detected it is less distinct than in the case of upstream. The time of the effect appearing downstream does not depend on traffic demand.

A drastic change, much like upstream occupancy, can be seen in Figure 2c, which shows upstream speeds. After the incident, the measured speed started to decrease sharply with the delay, depending on the traffic demand. As seen in Figure 2d, the downstream speed data cannot detect the effect of the incident.

The pattern of the upstream flow time series shown in Figure 2e is a surprising phenomenon. These time series do not show any difference, regardless of traffic demand, although the measurements appear noisy. The effect of the incident is much more prevalent in the downstream flow time series shown in Figure 2f. Here we can see a decrease in flow rate after the incident occurred. The bigger the traffic demand, the greater the decrease. This phenomenon confirms that due to a decrease in capacity, only part of the traffic demand can be adequately satisfied.

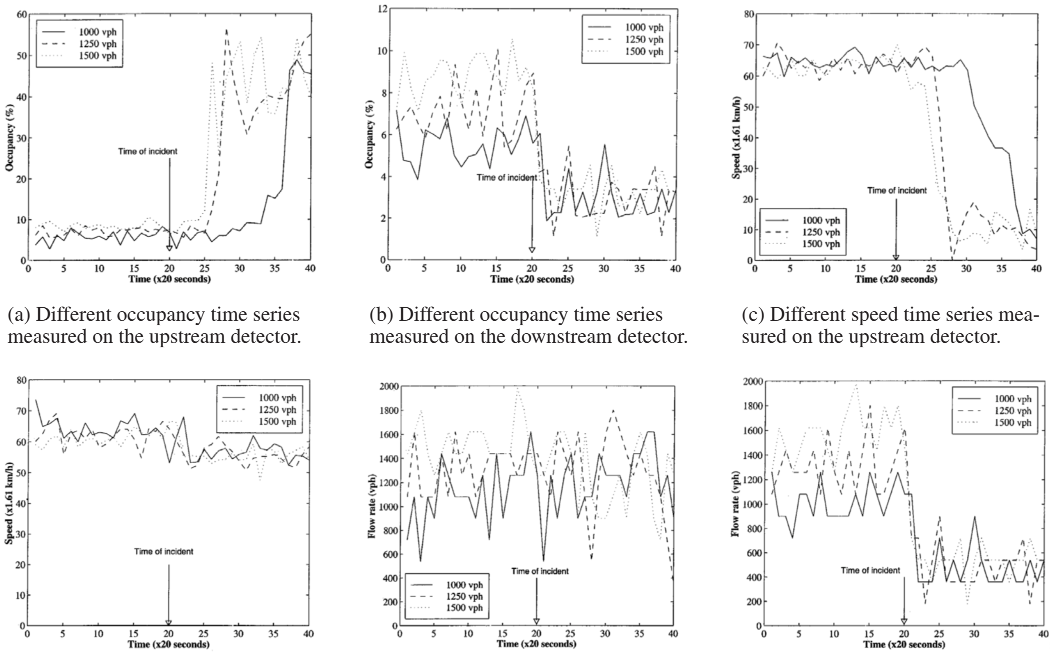
Of course, there is no guarantee that an occurrence of an incident will cause congestion. Incidents only cause problems on a given road segment if the current traffic demand is greater than the capacity of the road segment. For example, on a three-lane highway, a pulled over car is often not a problem even with higher traffic demands. Another example could be an accident at night on a three-lane highway that occupies only the outer lane. Although the capacity of the road segment is temporarily reduced, the effect can hardly be detected due to the minimal traffic demand at night. Another extreme is when an incident occurs on an already congested road segment. In this case, it is also not possible to detect an incident simply by taking the detector data into account.

B. Incident detection methods

An examination of the effects of incidents has shown that the occurrence of an incident distorts the time series of the traffic data in a way that is readily detectable. The challenge is that, depending on the type of incident, the capacity of the road network, the current traffic demand, and the distance between the detectors, the effect appears in the data at a different extent and delay. Because of this, for a truly reliable and rapid detection method it is necessary to carry out our studies on a dataset that is large and contains many different scenarios.

AID methods have been continuously published by researchers since the 1970s, but the area is still actively researched thanks to the spread of new machine learning methods. In this section, the significant AID methods from professional literature will be described. Since our own method is based on data from traffic detectors, we mainly focused on those methods that use detectors as data sources as well. Of course, methods based on other data sources will also described.

In professional literature, three main metrics are generally used to compare the performance of AID methods. The Detection Rate (DR) represents the ratio of correctly detected



(a) Different occupancy time series measured on the upstream detector.

(b) Different occupancy time series measured on the downstream detector.

(c) Different speed time series measured on the upstream detector.

(d) Different speed time series measured on the downstream detector.

(e) Different dow time series measured on the upstream detector.

(f) Different dow time series measured on the downstream detector.

Fig. 2: Time series measured on upstream and downstream detectors for different traffic demands [24].

incidents to the total number of incidents. Mean time to detect (MTTD) contains the average amount of time needed to detect incidents. False Alarm Rate (FAR) is the ratio of false incidents detected when no incident actually occurred. The metrics are detailed in Section IV-B.

To make the comparison of methods fair, we have implemented them wherever possible. It is important to point out that the performance of each method was measured on the new incident dataset described in our article, so that the operation of the methods is actually comparable, as each method had to recognize the same incidents. The results are summarized in Table III.

One of the best known of the early AID methods is the California algorithm [26], of which several modified versions have been made [27]. The method compares the occupancy values measured by two adjacent traffic detectors. The steps for comparison are shown in Figure 3, which is from one of the most frequently referenced: the 7th version of the algorithm. The 3 variable denotes the difference between the occupancy values measured on the detectors, the $OCCRDF$ variable denotes the ratio of the difference between occupancy values measured on the detectors, and the $DOCC$ variable denotes the occupancy value of the second detector in the direction of travel.

If these are larger than the pre-set threshold values $T1$, $T2$, $T3$, then the method considers the measurement to be an incident. Although the method is simple and surprisingly effective, the three thresholds are difficult to adjust. Setting these thresholds incorrectly and using noisy datasets can cause high FAR values. When comparing the results, the algorithm achieved 91.85% DR, 7.73% FAR and 7.28-minute MTTD values, which can be considered an average performance.

In order to reduce noise induced FAR, a Low-pass (LP) filter is used in the Minnesota algorithm [28], which is applied separately to the occupancy time series of the two detectors. The time series were examined with disjoint time intervals of 30 seconds. The operation of the algorithm is similar to that of the California algorithm: the steps and two thresholds defined by experts can be used to determine whether an incident had occurred at a given time. The disadvantage of the algorithm is that it cannot distinguish between congestion occurring because of a narrow cross section and actual incidents. Our studies also showed that using the parameter settings proposed in the article, although the DR value was high (99.25%) and only 2.2 minutes were measured for the MTTD value, the FAR value was extremely high at 48.23%.

Noise-induced difficulties are addressed by the University of California, Berkeley (UCB) algorithm [29] with a cumulative difference in occupancy values.

To do this, first the sum of the occupancy values measured so far on the two adjacent detectors is calculated separately, and then the difference between the two sums is taken. According to the authors, the change in the difference of the cumulative occupancy values follows Random-walk movement, so if the magnitude of movement rises above a pre-set threshold, their method identifies the given time as an incident. Our studies have shown that the method can achieve a low MTTD value of 3.34 minutes and a 3.4% FAR value, but the value of DR was only 82.22%, which is low compared to other methods.

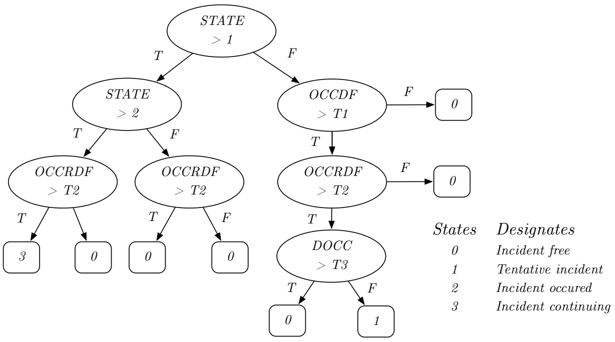


Fig. 3: The tree structure applied in the California algorithm #7

Other methods use a statistical approach instead of previous solutions. The algorithm of Levin et al [20] uses the value of the $OCCRDF$ variable from the California algorithm for detection. Based on the input dataset, the distribution of the incident and normal measurements are determined separately. In the case of a newly received measurement the Bayesian model is used to determine which distribution the measurement belongs to. Unfortunately, we were unable to reproduce the output of the method because we did not have access to the "Emergency patrol vehicle-assists database" to calculate the probabilities. This database is used to determine the likelihood that an incident will actually affect the capacity of the given road segment. Based on the studies published, the method achieved extremely good results on data with 30-second time intervals, but the Mean time to detect (MTTD) value was above 3.5 minutes.

In addition to the previous methods, there are also examples of techniques using different time series prediction for AID implementation in professional literature. In this approach, a prediction model is built for the examined traffic variable, and then monitoring the error of the predictions produced by the model. If this error is above a set threshold value then the given time can be considered an incident. Ahmed et al developed a Auto Regressive Integrated Moving Average (ARIMA) based method [30] that builds models on the occupancy time series of detectors. The 95% confidence interval of the forecast was used to determine the error limits. Using the confidence interval suggested in the article, we measured a MTTD of 0.98 minutes and a DR of 100%, which are good results, but a FAR value of 8.96% is considered high. This means almost every 10th alarm is false, which could be troubling for traffic management.

The RoadCast Incident Detection (RCID) [31] algorithm uses the Random Forest (RF) model to build a prediction model for each detector separately. It also incorporates data for holidays and events into the prediction model as an external data source. The integrating external data sources are described in detail in the article. The error limit for the predicted values are determined using the Quantile Random Tree Regression (QRTR) method. The QRTR method is a complement to the RF model, which estimates the range in which the prediction output will fall based on a known probability value (quantile). If the real measured values are outside the range defined by the quantile three times in a row, the RCID signals an incident. Unfortunately, due to the lack of external data sources, we were unable to reproduce their results, so we could not compare them with the other methods. In their own tests, they achieved very good results, typically with DR values above 80% and FAR values below 2%, but the MTTD values weren't published.

According to their studies, higher FAR values were measured in cases where an event was held in the area under investigation. In these cases, the forecasts themselves were inaccurate.

The DWT-Logit hybrid method [32] uses binary classification executed with logistic regression to implement AID. The output of the logistic regression indicates the probability of an incident. The probability from which it counts as an incident can be determined by setting a threshold value. The originally noisy data is cleaned with Discrete Wavelet Transformation (DWT), which is used to filter out high-frequency, probably noise-like components. The DWT-Logit hybrid method reached a DR value of 100% and an MTTD value of 1.19 minutes on our dataset, which is the best result of the examined methods. In contrast, the method's FAR value was extremely high at 27.04%, which is not acceptable in a real system.

In recent years, several publications have tried to achieve break through using Neural Network (NN). The authors of [24] treated the phenomena detected on upstream and downstream detectors separately, building separate Radial Basis Function Neural Network (RBFNN) models for the detectors. Different DWT coefficients were used as inputs for RBFNN models.

For upstream detectors, occupancy and speed coefficients were used, and for downstream detectors occupancy and flow coefficients were taken into account. Simulated and real data were both used for performance analysis. On the real dataset, which contained only 21 incidents, they reached a DR value of 95.2% and a FAR value of 0%, but the MTTD was not measured, which would have been important information.

Another study [33] focused on correctly setting the hyperparameters of the Neural Network (NN). The method uses fuzzy logic to set the topology of the hidden layers and the parameters of training for the NN. For setting the hidden layers, they used Stacked Auto Encoder (SAE) and Back Progapation (BP). In contrast, Li et al use the less common Extreme Machine Learning (EML) NN for incident detection [34]. The values of speed, flow and occupancy measured on the upstream and downstream detectors are used as inputs for a shared model. According to the author's tests, EML surpassed NN. Although neural network-based methods are promising and the results presented in the publications were better than other machine learning methods, unfortunately the results could not be reproduced in either case, as the publications lacked the precise hyperparameter settings.

In addition to neural networks, another commonly used method is the Support Vector Machine (SVM) [17], [18], [19]. In Motamed's dissertation [17] he developed an incident detection method with different SVM parameter settings and using data from two adjacent stations. The SVM model used flow, speed and historical speed values from the upstream detector and flow and speed values from the downstream detector as input parameters. In our comparison, their method reached a 100% DR value and a 2.76-minute MTTD value, but at the same time we measured a high FAR value of 12.84%.

Nowadays, in addition to traffic detectors, the GPS trajectories measured in vehicles are an increasingly widely used data source. This is often referred to as floating car data. The method developed by Ki et al [35] uses the phenomenon of high speed differences measured before and after the congestion. For detection they use a Feed Forward Neural Network (FFNN) method with layers (30,11,2). In their next work, they produced a modified version of the previous method [36], which uses layers (30,14,2).

In both cases [35], [36], the data for 3 road segments were examined: before, during, and after the congestion. A binary feature vector with 10 elements was defined for each of the three road segments based on the rate of speed changes measured. To do this, the set of possible values was divided into 10 disjoint ranges, and then the measured ratios were classified into these ranges. The value of an element of the feature vector was 1, where the measured ratio belonged, and 0 in the remaining 9. The binary feature vectors of the 3 road segments were added sequentially for the input of the neural network. The best of the published results were 77.3% DR and 8.6% FAR values. MTTD was not measured.

The Asakura method [23] used Travel Time (TT) instead of speed values for incident detection. The method compares current and previous TT values measured on a road segment in three steps. For each of the three steps a separate preset threshold value is defined, and if it is exceeded the algorithm flags it. Their best results were 50.2% DR, 0.015% FAR and 16.1 minutes MTTD values.

The advantage of methods based on GPS trajectories is that there is no need to build infrastructure and thereby serious installation and maintenance costs can be avoided. The disadvantage is that we do not know exactly how many vehicles on the road network we have information on. This value is called penetration. The literature showed that the results were worse than methods using detectors and, additionally, the reliability of the methods may be significantly reduced if the penetration rate is below 5%. Another problem with the use of GPS trajectories is that companies with large datasets do not make the stored data publicly available in many cases because of commercial or personal legal reasons, which further complicates research. However, trends show that with the development of technology and the spread of new vehicle communication devices [37], [38], this data source will give a new momentum to the research of AID methods.

Looking at the results in Table III, it can be said that the existing methods generally perform well. When examining the DR values, several methods [30], [32], [17] achieved the maximum performance of 100%, and measured relatively low MTTD values. Among the metrics, only the FAR values were too high, and none of the well-performing methods could reach a value below 5%. This is assumed to be due to the fact that models are more sensitive to noise in order to achieve high detection rates and fast detection.

III. Transient-based Automatic Incident Detection (TBAID)

In Section II-B, we saw that the methods currently available in professional literature have achieved good results according to the DR and MTTD metrics. However, it also turned out that the current methods are struggling to manage high FAR values, which can cause major problems for a traffic management center. False alarms can distract controllers from real incidents or cause them to lose focus. In addition, a lot of false alarms can disrupt the operation of the systems that build on the output of the algorithm. The Transient-based Automatic Incident Detection (TBAID) method that we have developed provides a solution to the problem of high FAR values while keeping the DR value high and the MTTD value low.

TBAID uses a new approach to reduce the number of false detections. One of the reasons for the high FAR values found in the current methods is that they want to detect the entire duration of the incidents, which can add extra noise to the training set. In this case, not only the occurrence of the incident, but also the permanent decrease in capacity may also be labeled as an incident, depending on the time at which the traffic authorities specified its end.

During our examination of incident behaviors (see Section II-A), we have noticed that the phenomena of transition between normal and incident states will occur in all cases and is difficult to confuse with other traffic phenomena. Conversely, there may be other reasons for a permanent capacity reduction. Therefore, TBAID focuses on detecting transient phenomena between normal and incident periods instead of the whole incident.

In order to achieve this, in addition to the previously known features describing the collected traffic data, we created new features that have not yet been used in professional literature, focusing specifically on detecting the transient phenomena. For the task of classifying non-incident and incident times, we applied the XGBoost (XGB) model [39], which has not yet been used in the AID area and, according to our studies, has further increased the accuracy of our method.

It is essential for machine learning methods to take into account as many useful features as possible during their operation. A feature is useful when it has sufficient discriminating power, so it can easily separate normal and incident times into two disjoint sets. These new features are produced by transforming raw Speed (SPD), Flow (FLW), and Occupancy (OCC) features measured on the detectors.

*A. Occupancy Difference (*OCCDF*) and Speed Difference (*SPDDF*)*

The analysis of the incident time series revealed that when an incident occurs, the difference between speed and occupancy values that differs from the average can be observed between upstream and downstream stations.

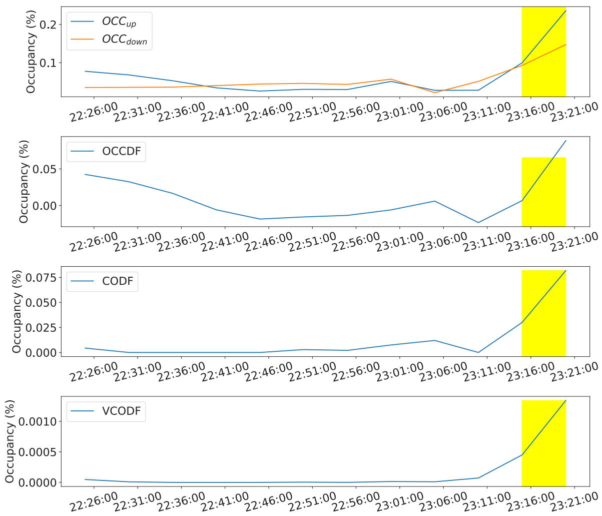


Fig. 4: Features related to occupancy data. The yellow rectangle indicates the presence of the incident.

The OCCDF and SPDDF features represent the differences between upstream and downstream detectors. At times when an incident occurs, the value of the OCCDF and SPDDF features will be higher than the average.

Denote the N long occupancy and speed time series measured on the upstream detector by $\mathrm{OCC}_{up} = \{occ_1, occ_2, \dots, occ_N\}$ and $\mathrm{SPD}_{up} = \{s_1, s_2, \dots, s_N\}$, and denote the N long occupancy and speed time series measured on the downstream detector by $\mathrm{OCC}_{down} = \{occ_1, occ_2, \dots, occ_N\}$ and $\mathrm{SPD}_{down} = \{s_1, s_2, \dots, s_N\}$.

The time series OCCDF$= \{occdf_1, occdf_2, \dots, occdf_N\}$ is a series of differences between OCC_{up} and OCC_{down}, the nth element of which is:

$$occdf_n = OCC_{up,n} - OCC_{down,n}, \quad n = 1, 2, \dots, N. \quad (1)$$

Similar to OCC, the time series SPDDF$= \{spddf_1, spddf_2, \dots, spddf_N\}$ is a series of differences between SPD_{up} and SPD_{down}, the nth element of which:

$$spddf_n = SPD_{up,n} - SPD_{down,n}, \quad n = 1, 2, \dots, N. \quad (2)$$

The second subfigure of Figure 4 shows an example of an OCCDF time series. The time interval affected by the incident was marked with yellow. Although the value of OCCDF increases during the period of the incident in line with the expected behavior, a similar phenomenon is observed at the beginning of the time series, which makes detection difficult and increases the FAR value.

B. Change of Occupancy Difference (CODF) *and Change of Speed Difference* (CSDF)

In order to reduce the effects of noise that interfere with detection, as mentioned in the introduction, we focus only on the occurrence of the incidents. During the examination of the incident data, OCCDF and SPDDF values significantly increased as the incident occurred. This observation was used to produce the CODF and CSDF features.

As a first step, the difference between the nth and $(n-1)$th element of the OCCDF time series is determined.

$$och_i = occdf_n - occdf_{n-1},$$

where $n = 2, 3, \dots, N$, $i = 1, 2, \dots N-1$ and $i = n+1$. Then in order to determine time series CODF$= \{codf_1, codf_2, \dots, codf_{N-1}\}$, all that is left to examine is whether the value of och_i is greater than zero:

$$codf_i = \begin{cases} 0 & \text{if } och_i < 0 \\ och_i & \text{otherwise.} \end{cases} \quad (3)$$

With this step, we can filter out negative changes that are not related to incidents. The third subfigure of Figure 4 illustrates the impact of this procedure. If we compare this to the second subfigure of Figure 4, which belongs to OCCDF, we can see that the disturbing noise has been significantly reduced.

As with occupancy, we can also define time series CSDF$= \{csdf_1, csdf_2, \dots, csdf_{N-1}\}$ in the case of speed, where:

$$csdf_i = \begin{cases} 0 & \text{if } sch_i < 0 \\ sch_i & \text{otherwise,} \end{cases} \quad (4)$$

while:

$$sch_i = spddf_n - spddf_{n-1}, \quad (5)$$

where $n = 2, 3, \dots, N$, $i = 1, 2, \dots N-1$ and $i = n+1$.

C. Variance of Change of Occupancy Difference (VCODF)

Although the CODF significantly reduced noise, our goal was to further clean the time series and highlight the phenomenon that is important to us. The time series Variance of Change of Occupancy Difference, $VCODF = \{vodf_1, vodf_2, \dots, vodf_{N-2}\}$ highlights significant changes in the CODF time series by defining sample variance between two consecutive elements. The calculation method for element number j $(j = i+1)$ of the time series VCODF is as follows:

$$vcodf_j = \left(codf_j - \frac{codf_j + codf_{j-1}}{2}\right)^2 + \\ + \left(codf_{j-1} - \frac{codf_j + codf_{j-1}}{2}\right)^2. \quad (6)$$

Figure 4 shows that VCODF reduced the noise in the time series compared to the CODF, while the period of the incident still stands out properly.

D. Difference to Typical Speed (DFTSPD)

When examining the speed data, another interesting feature is the deviation from historical speed, since the effects of the incident may result in a significant increase in the difference between current and historical data. Let the Typical Speed (TSPD)$= \{tspd_1, tspd_2, \dots, tspd_N\}$ be the time series of historical speeds. The nth element of an N long DFTSPD time series is:

$$dftspd_n = tspd_n - SPD_{up,n}, \quad n = 1, 2, \dots, N, \quad (7)$$

where $SPD_{up,n}$ is the nth element of the time series measured on the upstream detector, while $tspd_n$ is the nth element of the TSPD time series.

During our studies, we found that for the best results, data from the previous two weeks should be considered to determine historical behavior. Where no speed data were available, we used the speed limit for the given road segment.

E. Rolling window (WND) and features squared (SQRD)

In addition to the new features, we have defined two more optional transformation steps that further increase the reliability of our method.

When examining incidents, an important observation was that after the incident occurred, its effects would not appear immediately in the data. The reason for this is:

- the detectors are usually not installed evenly, so the distances between them vary,
- incidents occur in different positions compared to detectors,
- depending on the type of incident and current traffic, the effect will propagate at different speeds on the road network.

Since the impact of the incident may not appear immediately in the dataset and the magnitude of the effect also depends on the current demand, it is recommended to examine not only a point of time, but time windows in which the impact of the incident is more likely to be detected. To do this, the method we propose uses a fixed sized rolling time window. During training, we tried several time windows of different sizes, of which the 20-minute window size proved to be the best. The rolling window is denoted by WND.

Another optional transformation step is to square the values of the current features. By squaring, normal and incident data become further apart, thereby increasing the descriptive power of the available features, making it easier to detect incidents. Squaring is denoted by $SQRD$.

IV. EVALUATION

A. Dataset

During the evaluation, the Caltrans Performance Measurement System (PEMS) dataset [40] was used. The dataset is made up of measurements from approximately 39,000 measuring stations located along major routes in the state of California, USA, going back to 1999. The size of the dataset is currently about 12 terabytes, which is publicly available and free to download for anyone. Analyzing the total amount of that data would have taken too much time, so we only examined a subset of it. The evaluation was performed in District 3 (Sacramento area) for a one year period from January 1, 2016 to December 31, 2016 and the traffic data was collected by dual loop detectors. Traffic data in the PEMS dataset is available with 30-second and 5-minute aggregation, but the traffic data of 30-second aggregation was incomplete. Therefore, we were forced to use traffic data with 5-minute aggregation. The advantage of using the 5-minute data is that the data contains significantly less noise than the 30-second aggregation, but this increases the detection time because the effect of events is displayed with a higher delay.

We also have access to the incidents recorded by California Highway (CHP) via the PEMS Official Site [40]. We used this incident database as a starting point. As many incidents have no effect on traffic data, both automatic and manual filtering steps were required. In the first step, the incidents identified as accidents were selected, because they generally have an impact on traffic trends [12]. In the second step, we selected the incidents where there was a detector nearby and we received data from both detectors. In order to determine the position of the incidents and detectors, the absolute postmile value was used, which represents the distance in miles from the beginning of the road (or from the state border). In step three, we developed a graphical tool to display the 5-minute traffic data at the time of the incident to determine with the naked eye whether the incident caused a change in the measured data.

After the pre-filtering steps, we manually examined more than 5,000 incidents using our graphical tool. When selecting the incidents, we took into account whether, at the time of the incident, there was a visible difference from the historical behavior, a phenomenon described in Section II-A, shown on the detector pair. In many cases, there was no discernible difference between historical and incident data and none of the phenomena described in Section II-A were visible. This may be due to the fact that, depending on its severity, not all incidents have a real impact on the data, and it has also been observed that in case of a detector error, PEMS uses historical data to replace missing periods caused by the error, and therefore such incidents cannot be used. In the end, of the 5,000 incidents, 452 such cases were identified. A detailed description of these steps was published on the official page of our dataset [41].

In the evaluations, we used a randomly selected 60 percent of the incidents as a training dataset and the remaining 40 percent as a test dataset.

B. Metrics

The evaluation included the most frequently used metrics in professional literature: DR, FAR and MTTD.

Let $\mathcal{I} = \{I_1, I_2, \ldots, I_{|\mathcal{I}|}\}$ be the set of incidents in the dataset, and $\hat{\mathcal{I}} = \{\hat{I}_i | \hat{I}_i \in \mathcal{I},\ i = 1, 2, \ldots, |\hat{\mathcal{I}}|,\ |\hat{\mathcal{I}}| \leq |\mathcal{I}|\}$ be the set of detected incidents for which it is true that $\hat{\mathcal{I}} \subseteq \mathcal{I}$. The Detection Rate (DR) represents the ratio of correctly detected incidents to the total number of incidents that can be determined as follows:

$$DR[\%] = \frac{|\hat{\mathcal{I}}|}{|\mathcal{I}|} \cdot 100, \tag{8}$$

where $|\mathcal{I}|$ represents the number of incidents and $|\hat{\mathcal{I}}|$ represents the number of detected incidents.

The False Alarm Rate (FAR) shows the ratio of measurements incorrectly detected as incidents to the total number of non-incident measurements:

$$FAR[\%] = \frac{\#FDT}{\#NIT} \cdot 100, \tag{9}$$

TABLE I
THE FEATURES OF THE SCENARIOS USED IN THE EVALUATION.

Scenario	FLW	OCC	SPD	OCCDF	SPDDF	CODF	CSDF	VCODF	DFTSPD	WND	SQRD
#1	✓	✓	✓								
#2		✓	✓								
#3		✓	✓							✓	
#4				✓	✓						
#5				✓	✓					✓	
#6						✓	✓				
#7						✓	✓			✓	
#8					✓			✓	✓		
#9		✓	✓				✓	✓	✓	✓	
#10		✓	✓				✓	✓	✓	✓	✓

TABLE II
EVALUATION RESULTS, WHERE 95% DR, 8% FAR AND 8-MINUTE MTTD FILTERING CRITERIA WERE APPLIED.

Scenario	XGB			KNN			SVM		
	MTTD	DR	FAR	MTTD	DR	FAR	MTTD	DR	FAR
#1	4.38	94.25	3.29	6.43	95.40	3.76	3.13	96.67	6.93
#2	6.28	94.25	2.71	6.78	95.98	4.76	nan	nan	nan
#3	4.97	94.25	2.61	6.36	94.83	4.71	3.93	94.25	6.44
#4	5.67	94.25	6.19	7.64	94.25	5.04	3.58	97.78	7.99
#5	5.03	94.25	2.67	5.86	94.25	3.58	nan	nan	nan
#6	2.73	95.56	3.94	3.01	94.25	3.35	2.05	94.44	6.45
#7	4.53	94.25	1.71	6.23	94.83	3.63	3.63	94.25	5.79
#8	1.98	95.56	3.74	2.09	95.56	5.44	nan	nan	nan
#9	1.83	95.00	1.83	2.58	96.11	4.33	2.13	95.56	2.69
#10	2.13	95.56	0.93	nan	nan	nan	nan	nan	nan

where $\#FDT$ is the number measurements incorrectly detected as incidents and $\#NIT$ is the number of non-incident measurements. A measurement means a time interval, the size of which depends on the dataset.

Mean time to detect (MTTD) determines the average amount of time needed to detect incidents as follows. Let the time of occurrence of the detected incident $\hat{I}_i$ ($\hat{I}_i \in \hat{\mathcal{I}}$) be $\hat{I}_{i,T}$, and the time of detection be $\hat{I}_{i,DT}$. Based on this, the mean detection time of the incidents can be determined as follows:

$$MTTD[mins] = \frac{1}{|\hat{\mathcal{I}}|} \sum_{i=1}^{\hat{\mathcal{I}}} \hat{I}_{i,DT} - \hat{I}_{i,T}, \tag{10}$$

where $|\hat{\mathcal{I}}|$ represents the number of detected incidents.

C. Scenarios

Like the method presented by Motamed [17], we have defined several scenarios on the basis of which features have been taken into account. We examined several scenarios in order to compare the results and see if the new features we created actually improved the accuracy of the classification.

The defined scenarios are summarized in Table I. Each column of the table contains the names of the features (or transformations) we use, while each row contains a different scenario. A cell contains an ✓icon if the feature is present in the given scenario. Our goal in creating scenarios was to see the FAR decrease caused by the new features we defined.

D. Results

After determining the scenarios, in addition to the XGB model, we also trained K-Nearest Neighbor (KNN), SVM and Auto-Encoder Neural Network (AE-NN) classification models using the defined feature, to make sure XGB is indeed one of the best choices for Automatic Incident Detection (AID). It is important to note that the TBAID method ended up using the XGB model and scenario #10 (Table I).

The hyperparameters of the classification models were set using grid-search optimization, during which a significant number of hyperparameter settings were examined. During the evaluation, nearly 23,000 settings were evaluated to ensure a fair comparison of the results of the models and scenarios.

In evaluating the results, it was a challenge to manage the trade-off between the metrics. To do this, as a first step we set a threshold value for each of the three metrics, below which we did not accept the result.

Then in the second step, the results meeting the criteria were sorted based on the metrics in order of importance of FAR, MTTD, DR.

The results of the evaluation are summarized in Table II. In the case of results presented in Table 2, 95% DR, 8% FAR and 8-minute MTTD criteria were applied. These thresholds values can be considered strict, but the model-scenario pairs were generally above predetermined thresholds. With stricter settings, only the XGB model and more complex scenarios performed well. Each column of the table contains the metrics of the models, while the rows are the individual scenarios. In the event that a model-scenario pair did not meet the metric criteria, a "nan" value can be seen in the cell.

Although the evaluations were also carried out for the AE-NN model, unfortunately in the case of the vast majority we got "nan" values. For this reason, AE-NN is not included in Table II. This does not mean, of course, that AE-NN is a bad model, only that it did not work well with this domain and data.

As seen in Table II, from the point of view of DR none of the models stood out among the models used, as all of them performed similarly. The difference is in MTTD and FAR metrics. Looking at the two metrics, XGB performs visibly better than SVM and KNN models. This is particularly noticeable in scenarios where new features we have developed can be found. For the XGB model, we measured FAR values below 2% and MTTD values below 2 minutes several times. In scenario #10, we were able to reach a FAR of even 0.93%, but this resulted in an increase in MTTD value.

It is also worth noting that the values of the MTTD and FAR metrics are constantly improving as we move towards higher-numbered scenarios for every model. This clearly demonstrates that the features we defined can improve the output of MTTD and FAR metrics regardless of the model used.

The following stricter criteria were 97% DR, 2% FAR and 2-minute MTTD values. In this case, we no longer created a table, because we did not get evaluable results outside of the XGB - scenario #10 pair.

This also means that the use of the $SQRD$ transformation has actually improved the results, since the use of $SQRD$ is

TABLE III
METHODS FOUND IN PROFESSIONAL LITERATURE AND TBAID RESULTS ON THE DATASET WE PREPARED. THE TBAID METHOD USES THE XGB MODEL AND SCENARIO #10 (TABLE I).

Algorithm	DR (%)	FAR (%)	MTTD (mins)
California #7[26]	91.85	7.73	7.28
Minnesota[28]	99.25	48.23	2.2
UCB[29]	82.22	3.4	3.34
ARIMA-based[30]	100	5.24	1.30
DWT-Logit hybrid[32]	100	27.04	1.19
Motamed SVM[17]	100	12.84	2.76
TBAID	**97.22**	**1.56**	**1.89**

the only difference between scenarios #9 and #10. The XGB - scenario #10 pair achieved 97.22% DR, 1.56% FAR and 1.89-minute MTTD results, which are outstanding when compared to the results of the methods found in professional literature. Although the methods in Table III achieved up to 100% DR, at the same time very high results were measured for MTTD or FAR values. In contrast, TBAID was able to reach the lowest MTTD-FAR pair at 97% DR.

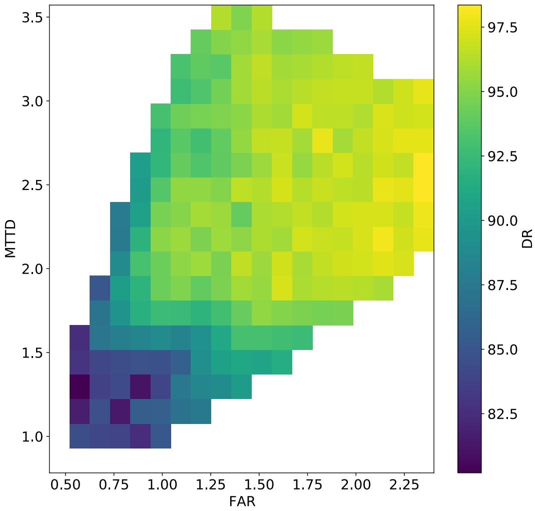


Fig. 5: Change of DR depending on the FAR and MTTD metrics.

We were also curious about the relationships between the change in values for each metric in the case of the XGB model and scenario #10. To examine this, we used the heatmap shown in Figure 5, in which we plotted DR as a function of FAR and MTTD values with the color scale indicated in the figure. The different results were obtained by changing the hyperparameters of the XGB model and the window size (WND).

Several correlations can be found in Figure 5. When seeking low MTTD and FAR values, the detection rate will also decrease significantly, only reaching values below 90%. If the MTTD is a less important metric, increasing the MTTD value allows for low, less than 1% FAR values along high DR values above 95%. It is also clear that an increase in FAR typically results in an increase in DR, but this behavior is not true for MTTD. According to Figure 5, the highest DR values can be measured around 2.5 minutes, but after that a slight decrease in DR values can be observed.

Overall, to determine the best result, it is necessary to define a system of criteria (such as the threshold-based solution we use) that determines how important each metric is to us. Generally, if we want to achieve an outstanding result for one of the metrics, it will have a negative effect on the output of the other metrics.

We also considered it important to examine the impact of the window size (WND) on the metrics. The results of the examination are summarized in the boxplots of Figure 6. We have examined a total of three window sizes: sizes 2, 4 and 6, which represent 10, 20 and 30 minutes.

The best MTTD and FAR values were clearly measured for window size 2, but at the same time the DR values were very weak, below 90%, so we do not recommend using window size 2. Conversely, in the case of window size 6 an outstanding DR value can be measured, but the results for MTTD and FAR values are too high. In general, the best results were achieved with window size 4 (20 minutes), which meant relatively low MTTD and FAR values in addition to an about 95% DR value.

V. CONCLUSION

One of the major problems in the transport of major cities is congestion on the road network, which is often caused by unexpected incidents. Quick and accurate detection of incidents can greatly reduce the negative effects they cause.

In this article, we presented the Transient-based Automatic Incident Detection (TBAID) method we developed, which utilizes an approach not yet used in professional literature to detect the occurrence of incidents.

To do this, we first created a new dataset using the PEMS database, which contains 452 incidents and their associated

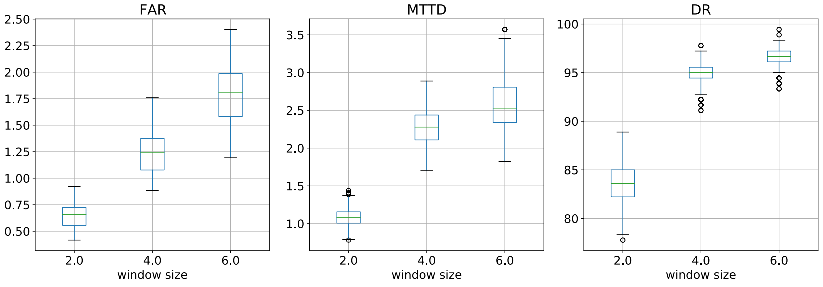


Fig. 6: The impact of window size on each metric.

traffic data. The dataset has been made publicly available for research purposes.

In addition to the new approach, new features and the XGB model were both used in the TBAID method. The operation of our method has been subject to detailed analysis, in which we have not only compared it with the methods found in professional literature, but also used other classification models in the evaluation of the results. For comparison, we used the standard DR, MTTD and FAR metrics. The TBAID method achieved outstanding 97.22% DR, 1.56% FAR and 1.89-minute MTTD values.

In the course of the evaluation, we have also put great emphasis on managing the trade-offs between the metrics and understanding how the performance of our method changes by setting different metric criteria.

References

[1] J. B. Bump, S. K. Reddiar, and A. Soucat, "When do governments support common goods for health? four cases on surveillance, traffic congestion, road safety, and air pollution," *Health Systems & Reform*, vol. 5, no. 4, pp. 293–306, 2019, doi: 10.1080/23288604.2019.1661212.

[2] N. Zhong, J. Cao, and Y. Wang, "Traffic congestion, ambient air pollution, and health: Evidence from driving restrictions in beijing," *Journal of the Association of Environmental and Resource Economists*, vol. 4, no. 3, pp. 821–856, 2017, doi: 10.1086/692115.

[3] O. K. Kurt, J. Zhang, and K. E. Pinkerton, "Pulmonary health effects of air pollution," *Current opinion in pulmonary medicine*, vol. 22, no. 2, p. 138, 2016, doi: 10.1097/MCP.0000000000000248.

[4] K. Chen, A. Schneider, J. Cyrys, K. Wolf, C. Meisinger, M. Heier, W. von Scheidt, B. Kuch, M. Pitz, A. Peters et al., "Hourly exposure to ultrafine particle metrics and the onset of myocardial infarction in augsburg, germany," *Environmental Health Perspectives*, vol. 128, no. 1, p. 017003, 2020, doi: 10.1289/EHP5478.

[5] X. Tian, H. Dai, Y. Geng, J. Wilson, R. Wu, Y. Xie, and H. Hao, "Economic impacts from pm2. 5 pollution-related health effects in china's road transport sector: A provincial-level analysis," *Environment international*, vol. 115, pp. 220–229, 2018, doi: 10.1016/j.envint.2018.03.030.

[6] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big data analytics in intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 383–398, 2018, doi: 10.1109/TITS.2018.2815678.

[7] J. Li, Y. Zhang, and Y. Chen, "A self-adaptive traffic light control system based on speed of vehicles," in *2016 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*. IEEE, 2016, pp. 382–388, doi: 10.1109/QRS-C.2016.58.

[8] R. W. Hall, "Non-recurrent congestion: How big is the problem? are traveler information systems the solution?" *Transportation Research Part C: Emerging Technologies*, vol. 1, no. 1, pp. 89–103, 1993, doi: 10.1016/0968-090X(93)90022-8.

[9] A. Skabardonis, P. Varaiya, and K. F. Petty, "Measuring recurrent and nonrecurrent traffic congestion," *Transportation Research Record*, vol. 1856, no. 1, pp. 118–124, 2003, doi: 10.3141/1856-12.

[10] J. Evans, B. Waterson, and A. Hamilton, "Evolution and future of urban road incident detection algorithms," *Journal of Transportation Engineering*, Part A: Systems, vol. 146, no. 6, p. 03120001, 2020, doi: 10.1061/JTEPBS.0000362.

[11] H. Zhang and A. Khattak, "What is the role of multiple secondary incidents in traffic operations?" *Journal of Transportation Engineering*, vol. 136, no. 11, pp. 986–997, 2010, doi: 10.1061/(ASCE)TE.1943-5436.0000164.

[12] N. Owens, A. Armstrong, P. Sullivan, C. Mitchell, D. Newton, R. Brewster, and T. Trego, "Traffic incident management handbook," Tech. Rep., 2010.

[13] T. Agenda, "Manual on uniform traffic control devices," 2017.

[14] R. J. Javid and R. J. Javid, "A framework for travel time variability analysis using urban traffic incident data," *IATSS research*, vol. 42, no. 1, pp. 30–38, 2018, doi: 10.1016/j.iatssr.2017.06.003.

[15] A. T. Hojati, L. Ferreira, P. Charles, M. R. bin Kabit et al., "Analysing freeway traffic-incident duration using an australian data set," *Road & Transport Research: A Journal of Australian and New Zealand Research and Practice*, vol. 21, no. 2, p. 19, 2012.

[16] Z. Chen and W. Fan, "Data analytics approach for travel time reliability pattern analysis and prediction," *Journal of Modern Transportation*, vol. 27, no. 4, pp. 250–265, 2019, doi: 10.1007/s40534-019-00195-6.

[17] M. Motamed et al., "Developing a real-time freeway incident detection model using machine learning techniques," Ph.D. dissertation, 2016.

[18] Y. Sun and Z. Hou, "A novel abnormal traffic incident detection method based on improved support vector machine," *Journal of Applied Science and Engineering*, vol. 21, no. 1, pp. 45–50, 2018, doi: 10.6180/jase.201803_21(1).0006.

[19] J. Xiao, "Svm and knn ensemble learning for traffic incident detection," *Physica A: Statistical Mechanics and its Applications*, vol. 517, pp. 29–35, 2019, doi: 10.1016/j.physa.2018.10.060.

[20] M. Levin and G. M. Krause, "Incident detection: A bayesian approach," *Transportation Research Record*, vol. 682, pp. 52–58, 1978.

[21] Y. Hernandez-Potiomkin, M. Saifuzzaman, E. Bert, R. Mena-Yedra, T. Djukic, and J. Casas, "Unsupervised incident detection model in urban and freeway networks," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2018, pp. 1763–1769, doi: 10.1109/ITSC.2018.8569642.

[22] E. D'Andrea and F. Marcelloni, "Detection of traffic congestion and incidents from gps trace analysis," *Expert Systems with Applications*, vol. 73, pp. 43–56, 2017, doi: 10.1016/j.eswa.2016.12.018.

[23] Y. Asakura, T. Kusakabe, L. X. Nguyen, and T. Ushiki, "Incident detection methods using probe vehicles with on-board gps equipment," *Transportation research part C: emerging technologies*, vol. 81, pp. 330–341, 2017, doi: 10.1016/j.trc.2016.11.023.

[24] A. Karim and H. Adeli, "Incident detection algorithm using wavelet energy representation of traffic patterns," *Journal of Transportation Engineering*, vol. 128, no. 3, pp. 232–242, 2002, doi: 10.1061/(ASCE)0733-947X(2002)128:3(232).

[25] N. H. Gartner, C. J. Messer, and A. Rathi, "Traffic flow theory-a state-of-the-art report: revised monograph on traffic flow theory," 2002.

[26] H. Payne, E. Helfenbein, and H. Knobel, "Development and testing of incident detection algorithms, volume 2: Research methodology and detailed results," Tech. Rep., 1976.

[27] M. Levin and G. M. Krause, "Incident-detection algorithms part 1. offline evaluation," *Transportation Research Record*, vol. 722, pp. 49–58, 1979.

[28] Y. J. Stephanedes and A. P. Chassiakos, "Application of filtering techniques for incident detection," *Journal of transportation engineering*, vol. 119, no. 1, pp. 13–26, 1993, doi: 10.1061/(ASCE)0733-947X(1993)119:1(13).

[29] W.-H. Lin and C. F. Daganzo, "A simple detection scheme for delay-inducing freeway incidents," T*ransportation Research Part A: Policy and Practice*, vol. 31, no. 2, pp. 141–155, 1997, doi: 10.1016/S0965-8564(96)00009-2.

[30] S. Ahmed and A. R. Cook, "Application of time-series analysis techniques to freeway incident detection," *Transportation Research Record*, vol. 841, pp. 19–21, 1982.

[31] J. Evans, B. Waterson, and A. Hamilton, "A random forest incident detection algorithm that incorporates contexts," *International Journal of Intelligent Transportation Systems Research*, pp. 1–13, 2019, doi: 10.1007/s13177-019-00194-1.

[32] S. Agarwal, P. Kachroo, and E. Regentova, "A hybrid model using logistic regression and wavelet transformation to detect traffic incidents," *Iatss Research*, vol. 40, no. 1, pp. 56–63, 2016, doi: 10.1016/j.iatssr.2016.06.001.

[33] C. El Hatri and J. Boumhidi, "Fuzzy deep learning based urban traffic incident detection," *Cognitive Systems Research*, vol. 50, pp. 206–213, 2018, doi: 10.1016/j.cogsys.2017.12.002.

[34] L. Li, X. Qu, J. Zhang, and B. Ran, "Traffic incident detection based on extreme machine learning," *Journal of Applied Science and Engineering*, vol. 20, no. 4, pp. 409–416, 2017, doi: 10.6180/jase.2017.20.4.01.

[35] Y.-K. Ki, N.-W. Heo, J.-W. Choi, G.-H. Ahn, and K.-S. Park, "An incident detection algorithm using artificial neural networks and traffic information," in *2018 Cybernetics & Informatics (K&I)*. IEEE, 2018, pp. 1–5, DOI: 10.1109/CYBERI.2018.8337551.

[36] Y.-K. Ki, W.-T. Jeong, H.-J. Kwon, and M.-R. Kim, "An algorithm for incident detection using artificial neural networks," in 2019 25th *Conference of Open Innovations Association (FRUCT)*. IEEE, 2019, pp. 162–167, DOI: 10.23919/FRUCT48121.2019.8981509.

[37] N. Varga, L. Bokor, A. Takács, J. Kovács, and L. Virág, "Anarchitecture proposal for v2x communication-centric traffic light controller systems," in *2017 15th International Conference on ITS Telecommunications (ITST)*. IEEE, 2017, pp. 1–7, DOI: 10.1109/ITST.2017.7972217.

[38] Á. Knapp, A. Wippelhauser, D. Magyar, and G. Gódor, "An overview of current and future vehicular communication technologies," *Periodica Polytechnica Transportation Engineering*, 2020, DOI: 10.3311/PPtr.15922.

[39] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794, DOI: 10.1145/2939672.2939785.

[40] "Caltrans pems," 2020, [Online; accessed 17-March-2020]. [Online]. Available: http://pems.dot.ca.gov/

[41] "Traffic incident dataset," 2020, [Online; accessed 14-September-2020]. [Online]. Available: https://gitlab.medianets.hu/anagy/incident_dataset

Attila M. Nagy received the B.S. and M.S. degrees in computer engineering from the Budapest University of Technology and Economics (BME), Budapest, in 2016. From 2016-2020, he was PhD student in computer engineering from the Budapest University of Technology and Economics (BME), Budapest. Since 2020 he is a Research Assistant in MEDIANETS laboratory at Department of Networked Systems and Services, BME. His research interests include time series data mining and analysis, traffic prediction, traffic congestion data analysis, automatic traffic incident detection.

Bernát Wiandt received his PhD from the Budapest University of Technology and Economics (BME) in 2017. Currently he is an Assistant Professor at the Department of Networked Systems and Services, Member of the Multimedia Networks and Services Laboratory.

He has done research on self-organizing systems, flocking and distributed task allocation, recently his research interests include machine learning and data analytics for smart cities and intelligent transportation management systems.

Vilmos Simon received his PhD from the Budapest University of Technology and Economics (BME) in 2009. Currently he is an Associate Professor at the Department of Networked Systems and Services, Head of the Multimedia Networks and Services Laboratory and Deputy Head of Department of Networked Systems and Services.

He has done research on mobility management and energy efficiency in mobile cellular systems and self-organized mobile networks, recently his research interests include machine learning and data analytics for smart cities and intelligent transportation management systems. He published 50+ papers in international journals and conferences, and acts as a reviewer or organizer for numerous scientific conferences. He serves currently as the Corporate liaison vice president for the Connected and Automated Mobility Cluster of Zala.

Developing a macroscopic model based on fuzzy cognitive map for road traffic flow simulation

Mehran Amini[1*], Miklos F. Hatwagner[2], Gergely Mikulai[3], and Laszlo T. Koczy[4]

***Abstract*—Fuzzy cognitive maps (FCM) have been broadly employed to analyze complex and decidedly uncertain systems in modeling, forecasting, decision making, etc. Road traffic flow is also notoriously known as a highly uncertain nonlinear and complex system. Even though applications of FCM in risk analysis have been presented in various engineering fields, this research aims at modeling road traffic flow based on macroscopic characteristics through FCM. Therefore, a simulation of variables involved with road traffic flow carried out through FCM reasoning on historical data collected from the e-toll dataset of Hungarian networks of freeways. The proposed FCM model is developed based on 58 selected freeway segments as the "concepts" of the FCM; moreover, a new inference rule for employing in FCM reasoning process along with its algorithms have been presented. The results illustrate FCM representation and computation of the real segments with their main road traffic-related characteristics that have reached an equilibrium point. Furthermore, a simulation of the road traffic flow by performing the analysis of customized scenarios is presented, through which macroscopic modeling objectives such as predicting future road traffic flow state, route guidance in various scenarios, freeway geometric characteristics indication, and effectual mobility can be evaluated.**

***Index Terms*—Fuzzy cognitive map, road traffic flow, macroscopic model**

I. Introduction

Traffic related issues have significant environmental, economic, and social consequences, including air pollution, the reduction of effectual mobility, the increase of fuel and time waste, etc. These problems can be mitigated to maintain citizens' safety, to balance the demand-capacity congestion ratio, and to reduce the cost related congestion through a wide range of methods from detecting frequent traffic congestions by using spatial congestion propagation patterns [1], to create intelligent traffic lights controller algorithms and cooperative scheduling [2]. Analyzing and modeling road traffic flow-associated parameters are the main aims of these methods. Modeling these parameters (e.g., density, time, velocity) is seen as indispensable to comprehend the heterogeneous behavior of road traffic [3], [4], even though it is difficult due to the nonlinearity and uncertainty caused by internal and external elements, for example, drivers' preferences, weather conditions, imprecision in the collected data by sensors [5]. Modeling these nonlinear and uncertain characteristics becomes more applicable by the development of intelligent transportation systems (ITS) and soft computing (SC) techniques [6].

The field of intelligent transportation systems arose in the early 1950s through the combination of multidisciplinary techniques such as information technology, electronics, and traffic engineering, in order to deal with transportation-related problems more efficiently by new data inference and communication tools [7]. Such systems mainly aim at enhancing the productivity of the current transportation systems in order to avoid traffic breakdowns and the traffic shifting from uncongested to a congested state. All these initiatives have similar characteristics; namely, first, they all seek to understand the essence of the road traffic flow at a particular location and then control its alterations. Thus, both rely mainly on conventional statistics-based approaches, e.g., Bayesian network models, nonparametric regression, history average, and autoregressive integrated moving average. These techniques are often unable to completely address the complexities associated with involved parameters of traffic and their relationships and mainly resulting in unreliable road traffic detection and prediction [8], [9].

By introducing self-learning data processing techniques rather than model-based estimation methods caused by the advancement in inferential intelligence, data-driven approaches have developed rapidly [10], [11]. The emphasis of the classical numeric methods is on assuming certain statistical behaviors of the system in advance, mainly based on stationary and deterministic features. Hence, they fail to model the complex, non-deterministic, uncertain behavior of the system, where intelligent self-learning data processing techniques could be able to model the complexity of the system on hand, based on understanding the available data to build up an adequate structure. This understanding of the system is achievable by sacrificing completeness and accuracy and by tolerating imprecision in order to attain tractability, cognition, and cost-effective solutions [12], [13]. Zadeh named the various methodologies based on intelligence and sub-symbolic representation of the phenomena 'Soft Computing' (SC) [14]. Recently, soft computing techniques such as fuzzy-based inference, neural networks, evolutionary and population-based computing, such as swarm intelligence, etc., have provided significant achievements in improving the performance of ITS. These enhancements are achieved mainly due to the massive changes in the data scale generated and collected from various sources by the involved stakeholders, e.g., governments,

[1,2] Department of Information Technology, Szechenyi Istvan University, Gyor, Hungary

[3] Doctoral School of Regional Sciences and Business Administration, Gyor, Hungary

[4] Department of Information Technology, Szechenyi Istvan University, Gyor, Hungary and Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Hungary

DOI: 10.36244/ICJ.2021.3.2

citizens, and the industry with respect to these systems [15], [16]. Intelligent transportation-based systems are indeed a well-suited area to apply soft computing techniques since the data provided here are full of uncertainty and vagueness, where technical disciplines of SC techniques such as approximate computing and randomized search can be properly employed [17], [18]. In previous studies, SC methods have been proposed in various transportation problems such as road traffic flow and state prediction in [8] and [17], vehicle route planning, and vehicular ad-hoc networks in [20]. Thus far, the abilities of SC-based techniques in terms of modeling the road traffic flow in networks of freeways have been more or less neglected.

Moreover, in traffic control engineering projects, the road traffic flow modeling has significant contributions, take for instance, strategy assessment and development for road traffic control management, the inspection, and forecast of road traffic conditions in dynamic networks in the short term, evaluating the effect of recent constructions and comparing alternatives, etc. [21], [22]. With regard to road traffic flow characterization, three classes representing three levels of the models have been applied: the macroscopic, microscopic, and mesoscopic levels [23]. At the macroscopic level, aggregate road traffic is modeled by global variables, i.e., velocity, density, and flow of the road traffic as a mass behavior, while individual vehicle behavior is considered at the microscopic level only [24]. Both aggregate and individual behaviors are analyzed at the intermediate mesoscopic level [25]. The first macroscopic road traffic flow model was introduced by Lighthill [26]; since then, these models have gained increasing attention because of their uncomplicatedness and low inferential complexity, the latter enabling real-time evaluation and actions. This study aims at introducing a new macroscopic model based on fuzzy cognitive maps as one of the SC techniques for networks of freeways simulation.

Kosko defined Fuzzy Cognitive Maps (FCM) as: “fuzzy feedback models of causality that combine aspects of fuzzy logic, neural networks, semantic networks, expert systems, and nonlinear dynamical systems” [27]. Since then, a wide range of FCM applications have been conducted, see i.e., [28], [29]. One of the most frequent applications of FCM is in describing and simulating systems, including uncertainty and imprecision [17], [30]. Although FCM applications in the risk analysis area based on the concepts of failure, incident, error, etc. have been proposed already [31], the research effort described here has primarily been to model uncertain and non-deterministic conditions of heterogeneous road traffic flow systems through developing a macroscopic level-based method. The proposed FCM model also leads to illustrating the key arguments supporting the approach based on FCM, i.e., the sophistication and the efficient computational effort. Accordingly, this paper is devoted to demonstrating the abilities of FCM in modeling road traffic flow based on historical data collected from the networks of freeways in Hungary. This approach will lead to predicting the future states of road traffic flow, some indications concerning the geometric and geographic characteristics of the freeways, and the overall behavior of the network in various road traffic scenarios.

The rest of the paper is outlined as follows. In the next section, various road traffic flow models along with an introduction of METANET and FCM as the basis of the proposed model are highlighted. The third section presents the proposed new method with implementation aspects. Following that, the description of the applied dataset is given and the steps of the proposed new algorithm are defined and elaborated. In the fourth section, the performance of the proposed model’s results is investigated. Some conclusions are presented in the fifth and last sections.

II. Models

The necessity of modeling road traffic flow was raised because of the importance of mathematically describing the dynamic and complex behavior of road traffic-related systems. The first theoretic model of road traffic flow was introduced in [32]. Since then, a variety of road traffic flow-based models with different properties have been proposed. These models have been developed for various aims ranging from system analysis and future state forecasting to the modification of the current infrastructures. Categorizing these models is mainly based on two factors, the level of details coupled with the differentiation between macroscopic, microscopic, and mesoscopic methods [33]. In this study, the model’s focus is narrowed down on discrete macroscopic characteristics, which lays emphasis on the overall behavior of vehicles over time. As well as the involved variables are discretized (both temporally and spatially) instead of using continuous variable, i.e., freeways are considered as a set of segments with defined lengths, and time is also divided into discrete intervals [34].

Subsequently, a generic integrated approach in Section Three is presented; as a matter of fact, the approach not only can be applied to modeling macroscopic road traffic flow, but it also illustrates the potential application of fuzzy cognitive maps in modeling complex and nonlinear systems, which are known notoriously as full of uncertainty and imprecision. This unified approach is presented by employing two particular models: METANET [35] from the class of macroscopic road traffic flow models and the fuzzy cognitive map approach [36] as a soft computing method through which recognizing, classifying, and modeling complex systems is a possible approach.

A. METANET

METANET was introduced as a program to simulate freeway networks in a macroscopic way [35]. This simulation of the road traffic behavior in networks of freeways is based on an overall road traffic flow modeling that was originally developed by Payne [37]. METANET, as the most recognized second-order macroscopic approach, has been used in engineering and control-related problems. Second-order approaches lay emphasis on vehicles density and velocity by characterizing them in dynamic equations [5]. These properties allow a reasonably low-time inferential process. Therefore, real-time network simulation and representation are efficiently possible. The freeway network is embodied by a directed graph, i.e., bifurcations and junctions are represented by the nodes of the graph, while the freeway sections between these places are characterized by the edges (links). A freeway with two directions is modeled as two distinct directed edges with reverse directions. Edges are assumed to possess homogeneous geometric properties, e.g., the number of lanes is fixed. On the other hand, heterogeneous freeways may be modeled by connected edges separated by nodes at the places where the change of geometry happens [35]. Nonlinear difference equations are reflected in the model to illustrate the evolution of the road traffic associated variables, i.e., average space-mean velocity v (km/h), average density ρ (veh/km/lane), and average flow q (veh/h).

In the METANET simulation, whenever the geometry of freeway changes, e.g., a lane rises or drops, a junction, etc., a node is added to the model. Connections among these nodes are called links. Afterward, links are separated into equal segments. The following are the essential equations that are employed for determining the road traffic variables for each segment i of link m [5].

$$q_{m,i}(k) = \lambda_m \rho_{m,i}(k) v_{m,i}(k) \tag{1}$$

$$\rho_{m,i}(k+1) = \rho_{m,i}(k) + \frac{T_s}{L_m \lambda_m}[q_{m,i-1}(k) - q_{m,i}(k)] \tag{2}$$

$$v_{m,i}\ (k+1) = v_{m,i}(k) + \frac{T_s}{\tau}[V[\rho_{m,i}(k)] - v_{m,i}(k) + \frac{T_s v_{m,i}(k)[v_{m,i-1}(k) - v_{m,i}(k)]}{L_m} - \frac{T_s \eta [\rho_{m,i+1}(k) - \rho_{m,i}(k)]}{\tau L_m(\rho_{m,i}(k) + \kappa)} \tag{3}$$

$$V[\rho_{\mathrm{m},i}(k)] = v_{free,m} \exp\,[-\frac{1}{b_{\mathrm{m}}}(\frac{\rho_{m,i}(k)}{\rho_{cr,m}})^{b_{\mathrm{m}}}] \tag{4}$$

where $q_{\mathrm{m},i}(k)$ represents the outflow of segment i in link m over the time frame $[kT_s, (k+1)\ T_s]$, $v_{m,i}(k)$ and $\rho_{\mathrm{m},i}(k)$, signify space-mean speed (average speed of vehicles passing a segment during a time period) and the density of segment i of link m at time frame k, respectively. L_m represent the lengths of the segments situated in link m, while λ_m is the number of lanes in link m, and T_s represents the simulation discrete time frame. In Eq. (3), τ, η, and κ are global variables with constant values for all links in the freeway. They are named time constant, anticipation constant, and model parameter, respectively. Additionally, $\rho_{cr,m}$ as critical density, b_m as the parameter of the fundamental diagram, and $v_{free,m}$ as free-flow speed are specific for the basic diagram of the computed link m [5], [35].

B. The Fuzzy Cognitive Map approach

As an advancement of classic cognitive maps [38], the concept of the fuzzy cognitive map was introduced by Kosko [36] for the purpose of dealing with shortcomings related to the binary nature of the original cognitive map model. FCMs integrate the model of cognitive maps, and the idea of fuzzy set proposed originally by Zadeh [39]. They contain fuzzy nodes or concepts to explain the non-binary states of the modeled system components (concepts) as well as the gradual intensities of causalities among them. Although both models are represented as directed and signed graphs, the causal mechanisms with imprecise causal data can be described adequately only by the FCM. In this approach, more human-like reasoning in complex dynamic systems is applied, both in the model structure and the related computational processes. A schematic illustration of the FCM is indicated in Fig. 1; connections and interrelationships among concepts are modeled with weighted arcs.

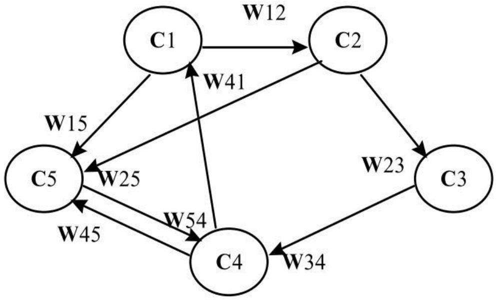


Fig. 1 A schematic illustration of simple FCM [7]

As it can be seen (Fig. 1), the variables of the system are represented by the indicated nodes C_1 to C_5. These variables are known as cause concepts where include nodes at the origin points of the arcs as well as effect concepts, where located at the terminal points of arcs. Take for instance, the C_1→C_2 connection, where C_1 is the cause variable because of impacting on C_2 as the effect variable. All concepts are individually identified by a number A_i commonly in the interval [0,1], which signifies its value in the model. Considering the signed (bipolar) fuzzy interval [-1,1] enables the model to assign grades or degrees of causality to the connections among the concepts [40]. The type of connection between two concepts signifies the influence of one concept (C_i) upon another one (C_j), where the interaction between them can be interpreted as excitatory or positive causality ($wij > 0$), and inhibitory or negative causality ($wij < 0$); and finally, null or no connection ($wij = 0$). Hence, the behavior of the system is warehoused and reflected in the structure of the concepts and the respective interconnections among them[41], [42].

Eq. (5) indicates the first introduced inference rule for the fuzzy cognitive map with $A^{(0)}$ as the initial activation vector; then the new activation vectors are computed at every individual step t and after defining the number of iterations after which the model will reach either its equilibrium point, or, the so-called limit cycle, or, eventually a chaotic behavior. The model shows these states under the following circumstances [36], [43], [44]:

- it stabilizes at fixed numerical values, achieving equilibrium at a fixed-point attractor with output values that are decimals in the interval.
- it displays limit cycle behavior, with output values falling into a loop of numerical values over a set time period.
- it illustrates a chaotic behavior, with each output value reaching a wide range of numerical values in a random, non-periodical, and non-deterministic manner.

Thus, updates are iteratively introduced until a terminal state has been reached. In this procedure, a state vector containing the activation degrees of the involved concepts is produced by the FCM at every discrete time frame [29].

$$A_i^{(t+1)} = f(\sum_{\substack{j=1\\ i\neq j}}^{n} w_{ji}\, A_j^{(t)}) \tag{5}$$

Although Eq. (5) had been employed in many FCM applications as the inference rule, a revised updating rule was introduced in [45], which is presented in Eq. (6), where the concept also considers its past value. The concepts are taken into account first, by their previous activation values, and second, by the activation values provided by other concepts and their corresponding weights.

$$A_i^{(t+1)} = f(\sum_{\substack{j=1\\ i\neq j}}^{n} w_{ji}\ A_j^{(t)} + A_i^{(t)}) \tag{6}$$

Various computation rules have been proposed [46], [47]. Applying the appropriate computational rule is determined by the type of the problem. Thus, the problem needs a profound understanding of all involved aspects before these rules are set. In both Eqs. (5) and (6), *f* is a threshold (squeezing) function. It expresses a monotonically non-decreasing function that defines the activation value of every concept toward the desired interval I, where either $I = [0, 1]$ or $I = [-1, 1]$, determined by the actual domain. The most broadly employed transfer functions are the bivalent, the saturation, or the trivalent, hyperbolic tangent, and sigmoid functions.

$$f(x) = \frac{1}{1 + e^{-\lambda(x-h)}} \tag{7}$$

The sigmoid function is given in Eq. (7). It is a continuous transfer function that provides an illimitable number of various states that are distributed within the desired hypercube. In the sigmoid transfer function, $\lambda > 0$ and h > 0 are user-defined parameters adjusting the function slope and offset, respectively. Greater values of λ raise the steepness, and it controls responsiveness to the variations of x. Fig. 2 shows the effect of the choice of value of λ in the transformation or inference results. Besides, increasing the activation value leads to the growth of the derivative [29].

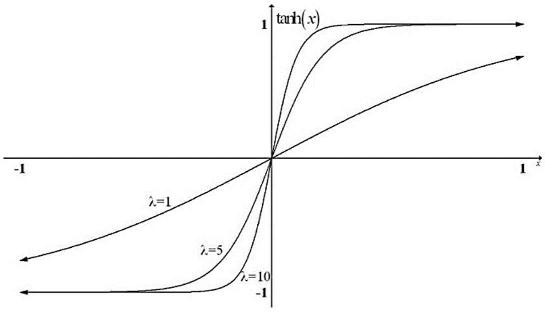


Fig. 2 Computation results determined by λ value [48]

III. THE PROPOSED NEW METHOD

In the sequel, an approach is presented which can integrate the macroscopic method with the fuzzy cognitive map approach to model road traffic flow. FCMs can be seen as recurrent neural networks with inference features, which include a set of neural computing entities or concepts [41]. Defining activation values for these concepts coupled with weight assignments is an essential part of creating the road traffic flow model based on FCM. In the proposed model, the activation values are assigned by an inference rule that is determined by combining the highlighted equations in Table 1. Therefore, it can be observed that the emphasis of the proposed integration is on two important factors; not only can activation values be computed by the values of the linked concepts with the corresponding causal weights at each time step, but concepts also take into account their own previous values. Algorithm steps will be elaborately explained in the implementation steps after describing the dataset.

TABLE I: INVOLVED INDICES/METHODS IN THE PROPOSED INFERENCE RULE

Author/s	Equation/Method	Usage
[35]	$\rho_{m,i}(t+1) = \rho_{m,i}(t) + \frac{T_s}{L_m\lambda_m}[q_{m,i-1}(t) - q_{m,i}(t)]$	Computing the future density of segment i (i.e., various sections with specific length between 100-18000 m) of link m (i.e., homogeneous freeway consist of several segments) at simulation time step.
[45]	$A_i^{(t+1)} = f(\sum_{\substack{j=1\\ i\neq j}}^{n} w_{ji}\ A_j^{(t)} + A_i^{(t)})$	Computing the value of concept C_i at time t, that the value of C_i is the calculated density in the segment (section) of the given link (freeway).

A. Data description

The presented FCM model was trained on road traffic data of the Hungarian network of freeways. Freeway users in Hungary experience complex and dynamic patterns of congestion. Besides other reasons, e.g., rather intensive road traffic caused by Hungary's strategic location in the European transport network and the system of corridors [49], this is mainly because of the increasing number of registered vehicles is Hungary, i.e., an increase of around 25% from 2010 to 2018 [50]. These problems lead to complex behavior with temporal and spatial alterations in road traffic. Therefore, modeling vehicles flow by available resources is seen as indispensable.

The dataset is collected from the online transaction processing server of the Hungarian e-toll system, which is an electronic system operated by the Hungarian national toll payment services for the whole network of motorways and primary highways of the country. This system enables the assistance and support of the verification of freeways usage, admittance, levying, and finally collecting the tolls of the standard road sections tollways [51]. The dataset contains seven variables: the name of the freeway, the section name (identifier), the collected e-toll over a span of one week in each section (segment) of the 212 freeway sections (links), which latter is considered as a proportional indicator of the number of vehicles, the time (per minute), the day, the length of the sections, and the number of the lanes in each section. These links include 2446 different segments altogether. Each segment length varies from 100 to 18,000 meters. For the sake of reducing the complexity of the model, a sample of 58 segments was selected, the full set of freeway sections through which Budapest is connected to the Austrian border.

Most of the road traffic models' aim is to explain the behavior of traffic-involved variables over the full range of operation, where identified locations have a pivotal role in the investigated dataset. This dataset can represent road traffic behavior based on location in a real-time manner. In Fig. 3, a sample of connections among three segments A, B, and C, are illustrated. Since the available dataset is based on time series, therefore, current traffic state in upstream segments can characterize road traffic flow conditions in downstream segments in the next time frames.

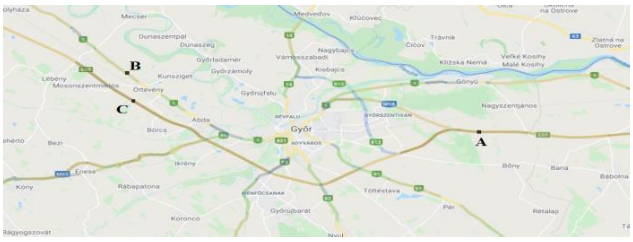


Fig. 3 A sample of freeway network and segment connections

The above-mentioned segments' causal relationships and correlations can be observed in Fig. 4. In the horizontal axis, the first digit represents the day, while the second and third digits represent the hour (in 24-hour format); an accurate behavior of road traffic flow over time is indicated, showing how traffic flow in the upstream segment can affect the subsequent segments. The calculated road traffic flow correlation among segments reveals that the correlation of A and B is 0.03, between A and C it is 0.9, and B and C correlate to 0.1; through which values various conclusions and correlation analyses can be conducted to identify the behavior and intensity of road traffic flow.

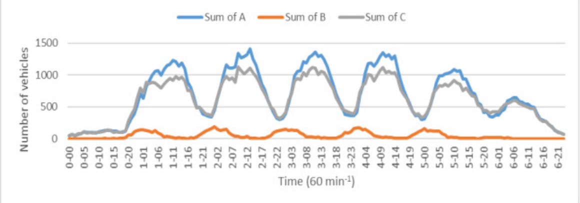


Fig. 4 Causal relationships of road traffic streamflow of three sample segments

B. The Model

The key deficit of applying FCM is the critical reliance on the initial expert judgment [52]. This issue stands out, particularly in modeling complex systems. In this research, extracted parameters of a macroscopic road traffic model have been applied for assigning initial values of concepts and weights. Each road segment is represented by a concept whose value is considered as the density ρ of segment i of link m, and the weighted arcs are set to a constant value based on $L_m\lambda_m$ variables as approximate capacity; where L_m denotes the length of the segments of link m, λ_m denotes the number of lanes of link *m*. The concepts and the weights initializations are set based on the aforementioned values. Afterward, the system is allowed to interact, and after every iteration, the new state vector is assigned newly generated values. This procedure will continue until the model exhibits an equilibrium state by reaching a stabilized condition at a fixed numerical boundary. The macroscopic road traffic model based on FCM is presented by the proposed algorithm in Fig. 5.

Simulation steps	
• Preprocessing:	
Step 1	Importing the dataset, consisting of seven columns: *m*, is the link name, *i*, is the segment name, $q_{m,i}$, is the no. of vehicles, T_s, the time day, L_m the segment length, and λ_m.the no. of segment lanes
Step 2	Calculating and adding the density (concept values) as the eighth column according to the equation: $$\rho_{m,i} = \frac{n_t}{L_m\lambda_m}$$
Step 3	Calculating $L_m\lambda_m$ as weight values initialization and adding the results as the ninth column to the dataset.
Step 4	Normalizing and adjusting concepts (density) and weights ($L_m\lambda_m$) in the [0,1] interval scale: $$X_{normalzed} = \frac{(X - X_{min})}{(X_{\max} - X_{min})}$$
Step 5	Defining connection weight matrix W_{ij} as $L_{mi}\lambda_{mi}$ based on the sequence of the segments, based on two possible types of causal relationships among concepts, i.e., when road traffic streams from concept C_i to C_j: • $W_{ij} > 0$ excitatory causality • $W_{ji} < 0$ inhibitory causality
Step 6	Transforming equation 1 to 2 for calculating the value of concept C_i as the density of segment *i* at time t: ($\rho_{m,i}^t$). 1) $A_i^{(t+1)} = f\left(\sum_{\substack{j=1\\ i\neq j}}^{n} W_{ij}\, A_j^t + A_i^t\right)$ 2) $\rho_{m,i}^{t+1} = f(\sum_{\substack{j=1\\ i\neq j}}^{n} \rho_{m,i,j}^t\ W_{ij} + \rho_{m,i,i}^t)$
• Main Algorithm:	
Step 7	Read the input initial concept state (input vector) A^0 as $\rho_{m,i}^0$.
Step 8	Define the relationship weight matrix W_{ij}
Step 9	Calculate the concept state $\rho_{m,i}^t$ according to the equation: $$\rho_{m,i}^t = f(\sum_{\substack{j=1\\ i\neq j}}^{n} \rho_{m,i,j}^{t-1} W_{ij} + \rho_{m,i,i}^{t-1})$$
Step 10	Apply the threshold function to output vector $\rho_{m,i}^t = f(\rho_{m,i}^t)$: $$f(x) = \frac{1}{1 + e^{-\lambda(x-h)}}$$

Step 11	If ($\rho_{m,i}^{t+1} = \rho_{m,i}^t$), stop;
	Else *Go To* Step 7;
	End

Fig. 5 Simulation steps of macroscopic road traffic flow model based on FCM

IV. RESULTS

Complex road traffic flow processes are characterized by various dimensions and components that are highly dependent and interconnected. For this reason, FCM as a soft computing technique is presented to address networks of freeways included imprecision and uncertainty. These uncertainties from the macroscopic modeling point of view are mainly connected with road traffic flow, density, and approximate capacity associated variables that can increase the probability of a breakdown and shifting the free flow state of traffic to congested flow [11], [34]. According to the applied algorithm in the previous section, segments of each link (freeway) are assigned as the concepts (nodes) of the FCM, where calculated density defines their values. In Fig. 6, a geographical representation of the selected segments is presented. There are 58 segments in the three investigated links, i.e., S_1, S_2, and S_3. Each of these links can be selected as the possible route from Budapest to the main Hungarian-Austrian border corridor represented as *E*. ES_1, and ES_2 are the endpoints of the S_1 and S_2 links, respectively.

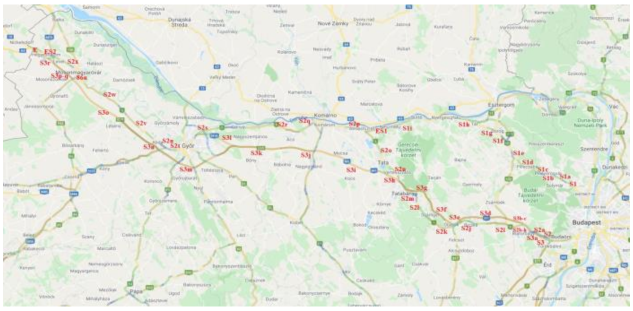


Fig. 6 Geographical locations of the selected segments

Therefore, the concept value initialization as the first step of FCM construction was determined corresponding to the real measured density, i.e., $\rho_{m,i}^{t}$ for segment (*i*), link (*m*), and time step (*t*). This value stands for downstream in the selected segment (transformed in the interval [0,1]), where the flow has not arrived at the next segment yet. Afterward, the FCM road traffic flow model developed by assigning weight values in accordance with the approximate capacity of each link through $L_m\lambda_m$. Fig. 7 depicts the initial state of the concepts and their respective interconnections along with the quantified weights in the interval [-1,1], which enables the classification of the degrees of causality among the segments. Two types of interactions among the segments are considered; where one segment (C_i) has excitatory causality on the subsequent segment (C_j), then $wij > 0$, which means downstream of C_i becomes upstream of C_j, while C_j has negative causality on C_i signified by a causal edge with a negative value from C_j to C_i . Consequently, the behavior of the segments coupled with interconnections among them is reflected and warehoused in the FCM thus constructed.

In Fig. 7, three alternative links that can be chosen from Budapest to the Austrian border are illustrated by S_1, S_2, and S_3 and their 58 nodes in the network. S_1 includes nine segments that end at segment ES_1 and joins to one of the S_2 segments; S_3, as the most chosen route, also has close interaction with the segments in S_2, which both end at segment *E* as the last Hungarian segment before entering Austrian territory. Greater activation values in the concepts (segments) are indicated by larger nodes in the modeled FCM; they represent greater density and show stronger activation values that cause greater impact on the network.

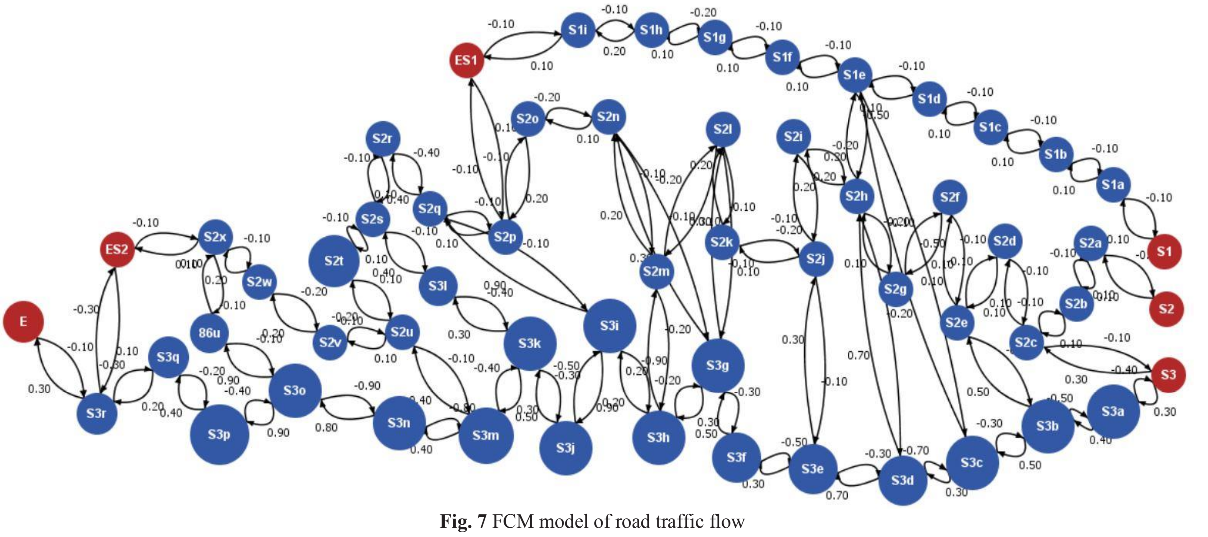


Fig. 7 FCM model of road traffic flow

In Fig. 7, an illustration of the FCM with initialized concepts and weights is shown. In terms of the initial state of the concepts, FCM begins to simulate the performance of the process. In every running step of the FCM, the state of concepts is computed according to step 9 (i.e., in the model simulation steps, see section 3). These steps are considered as those of a process in which the values of the defined concepts are analyzed. The value of every concept is assigned by considering all involved causal connection weights directed towards the concept and multiplying every weight by the value of the concept, which causes the connection, then adding the last value of every concept. In this simulation, a sigmoid function with $\lambda > 0$ was employed; therefore, the outcomes assumed values in the interval [0,1].

The FCM for road traffic flow modeling with initial vector values A_0 simulates the state of the system, and the values of the concepts for the desired iteration are illustrated in Fig. 8. The configuration of the FCM reasoning process was set on the stopping criterion when the fixed-point attractor was reached with 0.001 accuracy, and the values of concepts converged to the equilibrium region after seven iterations.

Table 2 presents the values of the concepts for these iterations. The results of the proposed FCM model of the road traffic flow follows a straightforward rule where each freeway segment is represented by a concept, and the density of each segment is signified by the computed values. As it can be observed, the values of these concepts mostly do not alter after the sixth step. Once the FCM reaches the equilibrium state, the new values of concepts are exchanged by the equivalent real values and vice versa. Evaluating the simulation results in the main segments of link S_3 as the most demanded part of the selected network illustrates the values of S_3, S_{3a}, S_{3c}, S_{3d}, S_{3e}, S_{3i}, S_{3j}, S_{3h}, S_{3k}, and S_{3n}, which formed the only group among all segments where after reaching a peak a downward trend followed. At the same time, the other five segments, namely, S_{3b}, S_{3f}, S_{3g}, S_{3l}, and S_{3m} showed constant increasing behavior. In this group of segments, when the FCM reaches the equilibrium point, the simulated density values of segments are transmitted to the real system and set the corresponding connected nodes. Finally, the FCM receives the simulated measurements from segments interactions, it interacts, then reaches an equilibrium point and transmits the density values of concepts to the whole model, and this iterative process continues.

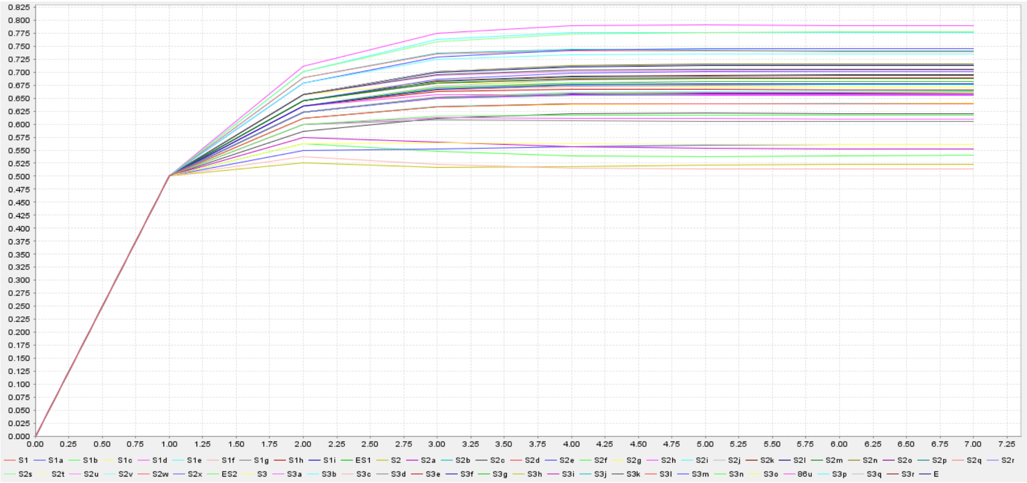


Fig. 8 Concepts' values variation in the FCM inference process

TABLE 2: Partial concepts' values of link S_3 in the first simulation steps

Step	S3	S3a	S3b	S3c	S3d	S3e	S3f	S3g	S3h	S3i	S3j	S3k	S3l	S3m	S3n
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
2	0.5622	0.5987	0.6225	0.5374	0.5987	0.6341	0.6225	0.6341	0.525	0.5744	0.69	0.5866	0.6457	0.5498	0.5622
3	0.5639	0.6121	0.6531	0.5225	0.6089	0.6629	0.6508	0.6702	0.5173	0.566	0.7371	0.6114	0.6795	0.5524	0.5476
4	0.5621	0.6118	0.6611	0.5151	0.6072	0.6668	0.6567	0.6796	0.5183	0.5566	0.7433	0.6195	0.6872	0.5569	0.539
5	0.5614	0.6106	0.6631	0.5135	0.6059	0.6665	0.6577	0.6818	0.521	0.5534	0.7424	0.6209	0.6891	0.5601	0.5382
6	0.5614	0.61	0.6635	0.5134	0.6056	0.6661	0.6577	0.6821	0.5225	0.553	0.7416	0.6207	0.6895	0.5611	0.5392
7	0.5614	0.6099	0.6635	0.5134	0.6056	0.6659	0.6577	0.6821	0.5229	0.553	0.7413	0.6204	0.6895	0.5611	0.5398

An advantage of the proposed FCM road traffic flow model is that it supports performing what-if simulation analysis based on altering the properties of the involved variables, and subsequently, it may be observed how the system behavior might be affected by changes in particular variables. As a most common congestion-related event in the freeways, lane drop is being caused by various possible events such as accidents, maintenance, etc., that can lead to shifting free-flow traffic to a congested state and delayed travel time. Therefore, this scenario was performed in the S_{3h} segment chosen as one of the high-density segments in the selected network, with a reproducible recurring congestion event potential forming a bottleneck location. Bottlenecks are parts of segments where traffic congestion is repeatedly evidenced, which possess a reducing capacity on the segment upstream and freely flowing traffic on downstream. They have two main types of dynamics, namely, slow-moving vehicles and frequent accidents, as well as static features, e.g., tunnel entrances [53], [54]. In this scenario, one of the two lanes of S_{3h} was dropped, and the FCM simulation process started. The results of this simulation can be observed in Table 3. As opposed to the first inference process, where values stop changing after the sixth step, in this case, the values of the concepts mostly continued changing after the sixth step as well.

Although one lane was dropped, the density in the S_{3h} segment decreased only slightly, which fact shows that density in the remaining lane dramatically increased. In the meanwhile, density value alterations in the corresponding connected nodes S_{3g} and S_{3i} with S_{3h} in link S_3 (S_{3g} and S_{3i} can be seen in Fig. 9) are indicated. Their geographical map and the FCM concept representations can be seen in Fig. 6, respectively. The simulation of one lane being dropped shows a downward trend up to 10% in the density of the subsequent connected segment (S_{3i}) compared to the results of the first simulation process in Table 2; whereas, it has a rising impact on the traffic state in the previous segments as the upstream of the sections S_{3h}, where S_{3g} and S_{3f} experienced upward trends in their density around 20% and 5%, respectively.

TABLE 2: ONE SEGMENT LANE DROPPED IMPACT ON THE NETWORK

Step	S3	S3a	S3b	S3c	S3d	S3e	S3f	S3g	S3h	S3i	S3j	S3k	S3l	S3m	S3n
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
2	0.5622	0.5987	0.6225	0.5374	0.5987	0.6341	0.6341	0.6341	0.5125	0.5744	0.69	0.5866	0.6457	0.5498	0.5622
3	0.5639	0.6121	0.6531	0.5225	0.6089	0.6616	0.6676	0.6725	0.4983	0.5654	0.7371	0.6114	0.6795	0.5524	0.5476
4	0.5621	0.6118	0.6611	0.5151	0.6073	0.6646	0.6752	0.6835	0.4971	0.5555	0.7432	0.6195	0.6872	0.5569	0.539
5	0.5614	0.6106	0.6631	0.5135	0.6061	0.664	0.6865	0.7263	0.4992	0.5321	0.7422	0.6209	0.6891	0.5601	0.5382
6	0.5614	0.61	0.6635	0.5133	0.6058	0.6634	0.691	0.8268	0.516	0.5105	0.7413	0.6207	0.6895	0.5611	0.5392
7	0.5614	0.6099	0.6635	0.5134	0.6058	0.6632	0.691	0.8269	0.5188	0.5015	0.7411	0.6204	0.6895	0.5611	0.5398

The presented simulations indicated the abilities of the FCM as a practical soft computing method, not only in macroscopic modeling to investigate the overall behavior of road traffic flow but also to capture the interesting and flexible features in terms of examining and monitoring alterations and modifications of the involved parameters that may affect the whole networks of freeways. These characteristics offer valuable information and can contribute to beneficial results related to the traffic engineering field, such as prediction and surveillance of the road traffic flow state in complex and uncertain networks, the estimation of the influence of new road constructions, or the comparison of various alternatives, the prediction of the effects of capacity increase or reduction, and the improvement and assessment of road traffic control associated strategies, detecting prone error locations and optimizing the network itself [5], [22], [35].

V. CONCLUSION

The current rapid progress in road traffic flow modeling urges a distinct emphasis on examining the capacities of various soft computing techniques in this field. This research paper proposed a novel macroscopic model of the road traffic flow, based on the FCM approach as one of the emerging soft computing techniques. Alongside being generic, as it can be adopted to most combinations of a macroscopic road traffic flow modeling, this approach introduced a new application of fuzzy cognitive maps in modeling a nonlinear and complex network of freeways for the very first time, with the focus on detecting the reasons of road traffic congestion. Also, sustainability-related objectives can be investigated with this approach as the key argument in designing and managing transportation systems, an approach that affects the potential of improving road traffic control strategies.

It is plausible that all contributions of a macroscopic road traffic flow model cannot be provided by the FCM model, mainly due to the problem complexity, and the obtained results may differ from the real state of the road traffic. However, any estimation technique can inherently include a tradeoff between model performance and operation speed. In this light, FCM provides real advantages; for example, once trained, the road traffic simulation can be performed rapidly and in most cases at an approved level of accuracy. Furthermore, the dataset of the study does not contain all segments that can affect road traffic behavior, but only those where the e-toll network is included. It is worth mentioning that the resolution of the representation of the networks of freeways can be dramatically improved by employing further mapping and data, consequently leading to more accurate – and obviously, more complex - FCM models with refined simulation results.

ACKNOWLEDGMENT

This research was supported by the National Office of Research, Development, and Innovation grant NKFIH K124055.

REFERENCES

[1] A. M. Nagy and V. Simon, "Traffic congestion propagation identification method in smart cities," *Infocommunications J.*, vol. 13, no. 1, pp. 45–57, 2021. DOI: 10.36244/ICJ.2021.1.6

[2] L. Alekszejenkó and T. Dobrowiecki, "Adapting it algorithms and protocols to an intelligent urban traffic control," *Infocommunications J.*, vol. 12, no. 2, pp. 57–62, 2020. DOI: 10.36244/ICJ.2020.2.8

[3] G. R. Timilsina and H. B. Dulal, "Urban Road Transportation Externalities: Costs and Choice of Policy Instruments," *World Bank Res. Obs.*, vol. 26, no. 1, pp. 162–191, Feb. 2011. DOI: 10.1093/wbro/lkq005

[4] W. Imran, Z. H. Khan, T. Aaron Gulliver, K. S. Khattak, and H. Nasir, "A macroscopic traffic model for heterogeneous flow," *Chinese J. Phys.*, vol. 63, no. December 2019, pp. 419–435, 2020. DOI: 10.1016/j.aej.2021.06.042

[5] S. K. Zegeye, B. De Schutter, J. Hellendoorn, E. A. Breunesse, and A. Hegyi, "Integrated macroscopic traffic flow, emission, and fuel consumption model for control purposes," *Transp. Res. Part C Emerg. Technol.*, vol. 31, pp. 158–171, 2013. DOI: 10.1016/j.trc.2013.01.002

[6] M. Amini, M. F. Hatwagner, G. C. Mikulai and L. T. Koczy, "An intelligent traffic congestion detection approach based on fuzzy inference system," 2021 IEEE 15th International Symposium on Applied Computational Intelligence and Informatics (SACI), 2021, pp. 97-104, DOI: 10.1109/SACI51354.2021.9465637.

[7] J. G. WARDROP, "ROAD PAPER. SOME THEORETICAL ASPECTS OF ROAD TRAFFIC RESEARCH.," *Proc. Inst. Civ. Eng.*, vol. 1, no. 3, pp. 325–362, 1952. DOI: 10.1680/ipeds.1952.11259

[8] M. Kalinic and J. M. Krisp, "Fuzzy inference approach in traffic congestion detection," *Ann. GIS*, vol. 25, no. 4, pp. 329–336, 2019. DOI: 10.1080/19475683.2019.1675760

[9] S. Ardabili, A. Mosavi, and A. R. Várkonyi-Kóczy, "Advances in machine learning modeling reviewing hybrid and ensemble methods," in *International Conference on Global Research and Education*, pp. 215–227, 2019. DOI: 10.1007/978-3-030-36841-8_21

[10] A. Csikos, Z. J. Viharos, K. B. Kis, T. Tettamanti, and I. Varga, "Traffic speed prediction method for urban networks- an ANN approach," in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, vol. 48, pp. 102–108, 2015. DOI: 10.1109/MTITS.2015.7223243

[11] P. Arnesen and O. A. Hjelkrem, "An estimator for traffic breakdown probability based on classification of transitional breakdown events," *Transp. Sci.*, vol. 52, no. 3, pp. 593–602, 2018. DOI: 10.1287/trsc.2017.0776

[12] W. Zhang et al., "State-of-the-art review of soft computing applications in underground excavations," *Geosci. Front.*, vol. 11, no. 4, pp. 1095–1106, 2020. DOI: 10.1016/j.gsf.2019.12.003

[13] Y. Yan, L. Wang, T. Wang, X. Wang, Y. Hu, and Q. Duan, "Application of soft computing techniques to multiphase flow measurement: A review," *Flow Meas. Instrum.*, vol. 60, no. November 2017, pp. 30–43, 2018. DOI: 10.1016/j.flowmeasinst.2018.02.017

[14] L. A. Zadeh, “Soft Computing and Fuzzy Logic,” *IEEE Softw*., vol. 11, no. 6, pp. 48–56, 1994. DOI: 10.1109/52.329401

[15] J. Zhang, F. Wang, K. Wang, W. Lin, X. Xu, and C. Chen, “Data-Driven Intelligent Transportation Systems: A Survey,” I*EEE Trans. Intell. Transp. Syst*., vol. 12, no. 4, pp. 1624–1639, Dec. 2011. DOI: 10.1109/TITS.2011.2158001

[16] E. O. Antonio D. Masegosa, Enrique Onieva, Pedro Lopez-Garcia, “Applications of Soft Computing in Intelligent Transportation Systems,” in *Soft Computing Based Optimization and Decision Models*, vol. 360, Springer International Publishing, pp. 153–175, 2018. DOI: 10.1007/978-3-319-64286-4_4

[17] R. Falcone, C. Lima, and E. Martinelli, “Soft computing techniques in structural and earthquake engineering: a literature review,” *Eng. Struct.*, vol. 207, no. November 2019, p. 110269, 2020. DOI: 10.1016/j.engstruct.2020.110269

[18] S. Nosratabadi, A. Mosavi, R. Keivani, S. Ardabili, and F. Aram, “State of the art survey of deep learning and machine learning models for smart cities and urban sustainability,” in *International Conference on Global Research and Education,* pp. 228–238, 2019. DOI: 10.1007/978-3-030-36841-8_22

[19] M. Lippi, M. Bertini, and P. Frasconi, “Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning,” *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 871–882, 2013. DOI: 10.1109/TITS.2013.2247040

[20] H. Hartenstein and L. P. Laberteaux, “A tutorial survey on vehicular ad hoc networks,” *IEEE Commun. Mag*., vol. 46, no. 6, pp. 164–171, 2008. DOI: 10.1109/MCOM.2008.4539481

[21] L. Zhang, Z. Yuan, L. Yang, and Z. Liu, “Recent developments in traffic flow modeling using macroscopic fundamental diagram,” *Transp. Rev.*, vol. 40, no. 4, pp. 529–550, 2020. DOI: 10.1080/01441647.2020.1738588

[22] S. Fulari, A. Thankappan, L. Vanajakshi, and S. Subramanian, “Traffic flow estimation at error prone locations using dynamic traffic flow modeling,” *Transp. Lett*., vol. 11, no. 1, pp. 43–53, 2019. DOI: 10.1080/19427867.2016.1271761

[23] S. P. Hoogendoorn and P. H. L. Bovy, “State-of-the-art of vehicular traffic flow modelling,” *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng*., vol. 215, no. 4, pp. 283–303, Jun. 2001. DOI: 10.1177/095965180121500402

[24] K. Nagel, P. Wagner, and R. Woesler, “Still flowing: Approaches to traffic flow and traffic jam modeling,” *Oper. Res.*, vol. 51, no. 5, pp. 681-710+837, 2003. DOI: 10.1287/opre.51.5.681.16755

[25] G. E. Cantarella, S. De Luca, M. Di Gangi, R. Di Pace, and S. Memoli, “Macroscopic vs. mesoscopic traffic flow models in signal setting design,” *2014 17th IEEE Int. Conf. Intell. Transp. Syst. ITSC 2014*, pp. 2221–2226, 2014. DOI: 10.1109/ITSC.2014.6958032

[26] M. H. Lighthill and G. B. Whitham, “II-A Theory of Traffic Flow on Long Crowded Roads,” *Spec. Rep.*, no. 79, p. 7, 1964. DOI: 10.1098/rspa.1955.0089

[27] M. Glykas, Fuzzy cognitive maps: *Advances in theory, methodologies, tools and applications*, vol. 247. Springer, 2010. DOI: 10.1007/978-3-642-03220-2

[28] A. Amirkhani, E. I. Papageorgiou, A. Mohseni, and M. R. Mosavi, “A review of fuzzy cognitive maps in medicine: Taxonomy, methods, and applications,” *Comput. Methods Programs Biomed.*, vol. 142, pp. 129–145, 2017. DOI: 10.1016/j.cmpb.2017.02.021

[29] G. Felix, G. Nápoles, R. Falcon, W. Froelich, K. Vanhoof, and R. Bello, “A review on methods and software for fuzzy cognitive maps,” *Artif. Intell. Rev*., vol. 52, no. 3, pp. 1707–1737, 2019. DOI: 10.1007/s10462-017-9575-1

[30] D. Pradeepkumar and V. Ravi, “Soft computing hybrids for FOREX rate prediction: A comprehensive review,” *Comput. Oper. Res*., vol. 99, pp. 262–284, 2018. DOI: 10.1016/j.cor.2018.05.020

[31] E. Bakhtavar, M. Valipour, S. Yousefi, R. Sadiq, and K. Hewage, “Fuzzy cognitive maps in systems risk analysis: a comprehensive review,” *Complex Intell. Syst.*, 2020. DOI: 10.1007/s40747-020-00228-2

[32] M. J. Lighthill and G. B. Whitham, “On kinematic waves II. A theory of traffic flow on long crowded roads,” *Proc. R. Soc. London. Ser. A. Math. Phys. Sci.*, vol. 229, no. 1178, pp. 317–345, 1955. https://www.jstor.org/stable/99769

[33] C. Pasquale, S. Sacone, S. Siri, and A. Ferrara, “Traffic control for freeway networks with sustainability-related objectives: Review and future challenges,” *Annu. Rev. Control*, vol. 48, pp. 312–324, 2019. DOI: 10.1016/j.arcontrol.2019.07.002

[34] A. Ferrara, S. Sacone, and S. Siri, “*First-order macroscopic traffic models*”, no. 9783319759593. 2018. DOI: 10.1007/978-3-319-75961-6_3

[35] A. Messmer and M. Papageorgiou, “METANET: a macroscopic simulation program for motorway networks,” *Traffic Eng. Control,* vol. 31, no. 8–9, pp. 466–470, 1990. https://www.researchgate.net/publication/282285780_METANET_a_macroscopic_simulation_program_for_motorway_networks

[36] B. Kosko, “Fuzzy cognitive maps,” *Int. J. Man. Mach. Stud*., vol. 24, no. 1, pp. 65–75, 1986. DOI: 10.1016/S0020-7373(86)80040-2

[37] H. J. Payne, “Freflo: a Macroscopic Simulation Model of Freeway Traffic.,” Transp. Res. Rec., no. 722, pp. 68–77, 1979.

[38] R. Axelrod, “*Structure of Decision: The Cognitive Maps of Political Elites*”. Princeton: Princeton University Press, 1976. https://www.jstor.org/stable/j.ctt13x0vw3

[39] L. A. Zadeh, “Fuzzy sets,” *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965. DOI: 10.1016/S0019-9958(65)90241-X

[40] C. D. Stylios and P. P. Groumpos, “Mathematical formulation of fuzzy cognitive maps,” *Proc. 7th Mediterr. Conf. Control Autom*., no. June 1999, pp. 2251–2261, 1999.

[41] G. Nápoles, M. L. Espinosa, I. Grau, and K. Vanhoof, “FCM Expert: Software Tool for Scenario Analysis and Pattern Classification Based on Fuzzy Cognitive Maps,” *Int. J. Artif. Intell. Tools*, vol. 27, no. 7, 2018. DOI: 10.1142/S0218213018600102

[42] G. Nápoles et al., “*Fuzzy Cognitive Modeling: Theoretical and Practical Considerations*”. Springer Singapore, 2019. DOI: 10.1007/978-981-13-8311-3_7

[43] B. Kosko, “Hidden patterns in combined and adaptive knowledge networks,” *Int. J. Approx. Reason.*, vol. 2, no. 4, pp. 377–393, 1988.

[44] I. Akgun, A. Kandakoglu, and A. F. Ozok, “Fuzzy integrated vulnerability assessment model for critical facilities in combating the terrorism,” *Expert Syst. Appl*., vol. 37, no. 5, pp. 3561–3573, 2010. DOI: 10.1016/j.eswa.2009.10.035

[45] C. D. Stylios and P. P. Groumpos, “Modeling Complex Systems Using Fuzzy Cognitive Maps,” *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*., vol. 34, no. 1, pp. 155–162, 2004. DOI: 10.1109/TSMCA.2003.818878

[46] E. I. Papageorgiou, “A new methodology for Decisions in Medical Informatics using fuzzy cognitive maps based on fuzzy rule-extraction techniques,” *Appl. Soft Comput.* J., vol. 11, no. 1, pp. 500– 513, 2011. DOI: 10.1016/j.asoc.2009.12.010

[47] P. P. Groumpos, “Large Scale Systems and Fuzzy Cognitive Maps: A critical overview of challenges and research opportunities,” *Annu. Rev. Control*, vol. 38, no. 1, pp. 93–102, 2014. DOI: 10.1016/j.arcontrol.2014.03.009

[48] J. L. Salmeron and C. Lopez, “Forecasting Risk Impact on ERP Maintenance with Augmented Fuzzy Cognitive Maps,” *IEEE Trans. Softw*. Eng., vol. 38, no. 2, pp. 439–452, 2012. DOI: 10.1109/TSE.2011.8

[49] F. T. László, T. Péter, “HUNGARY’S ITS NATIONAL REPORT,” *ITS national report*, 2018. [Online]. Available: https://ec.europa.eu/transport/sites/transport/files/2018_hu_its_progress_report_2017.pdf.

[50] E. C. E. UNECE, “EU transport in figures - Statistical Pocketbook 2020, Number of registered passenger cars in Hungary from 1990 to 2018,” *European Commission*, 2020.

[51] GSMPRO Kft.,“General Terms and Conditions,” *https://tracker.gsm-pro.hu/. 2019*.

[52] E. I. Papageorgiou and P. P. Groumpos, “A weight adaptation method for fuzzy cognitive map learning,” *Soft Comput.*, vol. 9, no. 11, pp. 846–857, 2005. DOI: 10.1007/s00500-004-0426-z

[53] R. L. Bertini and M. T. Leal, “Empirical study of traffic features at a freeway lane drop,” *J. Transp. Eng*., vol. 131, no. 6, pp. 397–407, 2005. DOI: 10.1061/(ASCE)0733-947X(2005)131:6(397)

[54] S. M. S. Seliman, A. W. Sadek, and Q. He, “Automated Vehicle Control at Freeway Lane-drops: a Deep Reinforcement Learning Approach,” J. Big Data Anal. Transp., vol. 2, no. 2, pp. 147–166, 2020. DOI: 10.1007/s42421-020-00021-0

Mehran Amini is a Ph.D. candidate in computer science in the Department of Information Technology at Széchenyi István University, Győr, Hungary. He has almost a decade of professional data analysis expertise, primarily in business intelligence. Computational intelligence and machine learning algorithms in modeling complex systems and risk analysis are among his main research interests. He also teaches Bioinformatics and IT project management.

Miklos F. Hatwagner is an Associate Professor in the Department of Information Technology at Széchenyi István University, Győr, Hungary. He holds a Ph.D. in Information Science from Széchenyi István University (September 2013). He has been working for over ten years as a Researcher in several research projects related to the development of novel parallel implementations of various evolutionary algorithms, the effective error handling techniques in distributed environments, and their application. He has been involved in several national research projects. He was also a member of the Hungarian ENUM project team. Later he turned his attention to the Fuzzy Cognitive Maps (FCM), the training and application of them to solve several problems arose in the fields of management, environmental protection, etc. He is the author or co-author of approx. 50 conference or journal papers. He has over 100 citations from independent researchers (h-index = 8 in Google Scholar and hindex = 7 in Scopus). His research interests include evolutionary algorithms, optimization, parallel computing, info-communication, Fuzzy Cognitive Maps, decision support, machine learning.

Gergely Cs. Mikulai received the B.Sc. degree in Mechanical Engineering at Budapest University of Technology and Economics (BME) in 2016. He received an M.Sc. degree in Business Development at Óbuda University in 2018. He is currently with Ph.D. Programme of Regional and Economic Sciences with Transdisciplinarity focus. His research interests mainly include route selection issues, using mostly fuzzy signature rule base evaluation.

Laszlo T. Koczy received the M.Sc., M.Phil. and Ph.D. degrees from the Technical University of Budapest (BME) in 1975, 1976, and 1977, respectively; and the D.Sc. degree from the Hungarian Academy of Science in 1998. He spent his career at BME until 2001 and from 2002 at Szechenyi Istvan University (Gyor, SZE). He has been a visiting professor in Australia, Japan, Korea, Austria, Italy, etc. His research interests are fuzzy systems, evolutionary and memetic algorithms, and neural networks, as well as applications in infocommunications, logistics, management, and others. In the last years, he has focused on NP-complete problems, especially route selection and optimization and the application of metaheuristics for approximate solution of such complex tasks. He has published over 775 articles, most of those being refereed papers, and several text books on the subject. His Hirsch-index is 40 by Google Scholar (based on 7300 citations there).

QoS Impacts of Slice Traffic Limitation

Khalil Mebarkia, and Zoltán Zsóka

***Abstract*—Slicing is an essential building block of 5G networks and beyond. Different slices mean sets of traffic demands with different requirements, which need to be served over separated or shared network resources. Various service chaining methods applied to support slicing lead to different network load patterns, impacting the QoS experienced by the traffic. In this paper, we analyze QoS properties applying a theoretical model. We also suggest appropriate parameter setting policies in slice-aware service function chaining (SFC) algorithms to increase QoS. We evaluate several metrics in different analysis scenarios to show the advantages of the slice-aware approach.**

***Index Terms*—Network Slicing, QoS, SFC, VNF, 5G Networks**

I. Introduction

Emerging communication technologies like 5G allow the provision of services with extended requirements. For example, new application sets can be served, which might combine high data rates, low latency, and extended reliability needs to be satisfied by the network.

Besides the progress in the radio networking part, which allows higher access and data rate for the clients, the control and data plane handling in the core part include innovative solutions. One of these is the support of Virtual Network Functions(VNFs), a technique for distributing the elaboration steps of traffic among some nodes instead of loading only central ones. VNF-capable nodes allow to start/scale/stop elaboration functions in virtual machines realized through various virtualization techniques.

The building blocks and management architecture of VNF-based solutions are described by a standard of the European Telecommunications Standards Institute (ETSI) [1]. Typical functions to be virtualized are Firewall, balancing, compression, and shaping of the traffic load. If a series of VNFs have to be considered in the provision of a service request, the task of Service Function Chaining (SFC) has to be performed to select appropriate VNF-capable nodes. Then, the traffic should pass through this chain of serving nodes to get the required elaborating functions.

A further novel concept introduced in 5G is *slicing*. It allows the definition of multiple service sets and a set of networking or even infrastructure resources to serve their requests. Fig. 1 illustrate this concept.

Since slices represent different types of traffic with different statistical properties and quality requirements, the requests belonging to them need appropriate handling with special SFC and routing solutions. The service chains apply both functional and networking resources as VNF-capable nodes and transport links, respectively. At the same time, the network-related part of Quality of Service (QoS) needs can be satisfied only with settings in the networking devices.

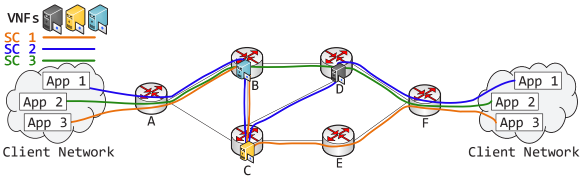


Fig. 1. Slice Concept of different services in different slices

The task of providing slices is twofold. On the one hand, higher-level problems as request admission, VNF resource selection, orchestration, or pricing and billing require intelligent control plane functions. On the other hand, we have problems for the lower level, like assignment of VNF and network resources and adjust settings in devices for handling QoS requirements of slices.

In this paper, we analyze slice-aware SFC mechanisms from the network QoS point of view. We propose different policies for adjusting the SFC parameters with the queue-level settings and evaluate their behavior. We consider only the VNF functional capability of the nodes and neglect their exact resource limits. Our concept concentrates on using network resources, and we aim to preserve them for other slices to hold the QoS expectations.

The paper is structured as follows. Section II summarizes the results of related works. In Section III we present methods allowing QoS in slicing, and we define the most important analysis metrics. In Section IV we propose policies for parameter setting in QoS-aware Service Chaining algorithms. We analyze the policies in Section V We conclude the paper in Section VI.

II. Related works

Papers [2], [3] address NFV as a promising architecture proposed to increase the scalability and the functionality of the network by leveraging virtualization technologies. It describes how telecommunication networks and services are designed and operated when traditional Network Functions (NFs) get transformed into VNFs.

Different approaches have been proposed by industry players, research institutes, and mobile operators to standardize SFC. In [1], ETSI defines a network service as a chain of VNFs. It emphasizes the demand for a new set of orchestration and management functions.

ETSI defines SDN usage in an NFV as an architectural framework and proposes a framework with three main components: VNFs and two subsystems termed respectively Manage-

K. Mebarkia and Z. Zsóka are with the Department of Networked Systems and Services, Budapest University of Technology and Economics, 1117 Budapest, Magyar tudósok körútja 2, Hungary (e-mail: mebarkia@hit.bme.hu; zsoka@hit.bme.hu).

DOI: 10.36244/ICJ.2021.3.3

ment and Orchestration (MANO) and Network Function Virtualization Infrastructure (NFVI) where VNFs are deployed. While in the RFC 7665 [4], IETF defines the service function chain as an ordered set of abstract service functions and ordering constraints. IETF also describes an SDN based SFC architecture. In this, an SFC classifier in the data plane performs a classification of end-to-end traffic to determine which VNF should be chained to process the traffic based on its requirements.

Various scientific works address the placement and chaining of VNFs. The proposed solutions focus on selecting the proper nodes for deploying a VNF for a specific traffic demand, or solve the SFC problem assuming the network topology and the demands, with VNF capabilities and VNF requirements, respectively. For instance, the authors in [5] propose a placement algorithm, which takes into consideration hardware accelerator resources in addition to compute resources. They aim to optimize the use of resources in Network Function Virtualization Infrastructure (NFVI), while placement algorithms must consider the presence of accelerators in NFVI nodes. They describe an Integer Linear Programming (ILP) for the accelerator-aware VNF placement problem. In [6], the authors study the VNF placement problem in SDN/NFV-enabled networks. They formulate the problem as a Binary Integer Programming (BIP) in which they aim to minimize a weighted cost, including the VNF placement cost. The authors propose a Double Deep Q Network-based VNF Placement Algorithm (DDQN-VNFPA) using deep reinforcement learning.

Other papers address similar problems while considering multiple slices in the network. Although Network Slicing is one of the most crucial parts of 5G core networks, its definition has never been unique, clear, and precise. It is varying from different perspectives of the various service providers. For instance, Next Generation Mobile Networks (NGMN) [7] defines a network slice as a set of network services that consists of 3 layers, Service Instance Layer, Network Slice Instance Layer, and Resource Layer. The network slice runs on top of physical resources where network services and resources conform to a logical network to deliver specific requirements. Slice can also be defined as a set of network and VNF resources, which can support one or more services, each with a prescribed series of VNFs that the service traffic shall pass. The supported services can be told to *be in the slice*.

The authors of [8] formulate the problem of statically embedding service chains into slices while also considering network link capacities. In [9] a Mixed Integer Linear Program (MILP) formulation is given for the problem of optimizing slices over multiple domains and accepting multiple services in each slice. The authors also present a heuristic that can guarantee the QoS requirements for the services by allocating the needed resources for the slices. Another work on the optimal allocation of VNF resources in 5G networks with cross-domain slices is [10], which introduces an ILP formulation and a Multi-layer based Knap-sack-based heuristic. Their solution aims to minimize the number of VNFs hosting the functions that constitute different network slices. Various QoS metrics are taken into account while the slices' set gets reorganized each time a service request arrives.

The authors in [11] present models for sliced networks, and investigate the cost reduction promises of using the NFV and network slicing technologies. In the models the slice deployment costs are allocated to show the network efficiency with slicing, while considering one specific demand that is realized as a service consisting of chained VNFs.

The authors in [12] focus on the end-to-end life-cycle management of network slices on different sites using a single management and orchestration entity with a coherent proof of concept. They propose algorithms for efficiently activating, deactivating, and decommissioning the network slices, using real-time status information from Network Slice Management Function (NSMF). The results show that by adopting a better strategy in these algorithms for controlling various phases of the slice life-cycle, the response time can be reduced for a user request by 50%.

Paper [13] presents a survey of works on slice admission control, citing and grouping works of various methodology. It presents multiple objectives for admission control, from revenue optimization to fairness, and mentions the priority-based strategy. Note that instead of admission priorities, our paper speaks about packet service priorities in the network queues, which is a different aspect.

The authors of [14] consider slices with demands with uncertainty in their number and requested resources. Their model involves a probabilistic approach of provisioning the node and link resources to fulfill the requirements. The problem of mapping the uncertain demands on the resources is formulated as a nonlinear constrained optimization problem, and then it is reduced to a parameterized a mixed integer linear programming (MILP) problems. The consideration of the uncertainty allows mappings that might be used in dynamic scenarios too.

Paper [15] extends the slice demand mapping problem to include also guarantees on the end-to-end latency of the traffic and to use a combined objective for the optimization. The authors consider the option of flexible routing, or in other words, load balancing of traffic via multiple paths, and present a mixed binary linear program (MBLP) formulation for the problem. In their model, the latency calculation considers only the propagation delay and a static NFV delay while neglecting the queueing delays at the network nodes. Due to the high complexity of the full formulation of the problem, a reduced formulation is contributed too. The slice mapping solution presented in [16] also applies multiple paths, but for a different reason. The paper takes under the scope another important requirement for SFC, and design slices with guaranteed availability.

In [17], the service chaining problem is considered in a two-layer model, which consists of a Functional Layer (FL)and a Network Layer (NL). We logically separate the topology of VNF-capable nodes and functional links allowed among them and the topology of network nodes and links. We address the problem of considering the current load state of both the functional links and the network below it. We discuss how to determine the SC according to the required bandwidth and VNF order while avoiding overloads on the network links. As a result, we propose heuristic and ILP solutions to formulate these challenges. These solutions are based on the dynamic calculation of SC by considering the current network load to avoid the use of heavily loaded links. The heuristic algorithm *OdAASP* (Overload Avoiding Augmented Shortest Path) determines the shortest path between source and destination with the awareness of considering overload avoidance. As a comparative solution, we use the algorithm *SFC-CSP* (SFC-Constrained Shortest Path) that finds the shortest path and satisfies a given SFC constraint as proposed in [18].

In [19], we introduce heuristic service chaining solutions that consider shared slicing and apply a kind of preservation of network resources for other slices to hold the QoS expectations. The application of these solutions can control the network link loads in several situations. Our objective now is to discuss systematically the concept of slicing-awareness by resource preservation, and to extend the analysis to more detailed QoS properties. Our aim is to show the importance of handle network slicing considering shared resources and dynamic service requests, to ensure the low latency and guaranteed bandwidth for different services. Our experience is that this topic is not well discussed in the state-of-the-art works.

III. QoS in slicing

Slices are assumed to be service sets allowing multiple requests and using a determined set of network and VNF resources. [8] defines the sharing property for VNFs. This value describes how the available VNF resources can be shared among slices. Since we concentrate on the network resources, the model is extended to links and simplified to the two basic cases: fully shared resources and lack of sharing. We consider the network as a two-layered system:

FL the Functional Layer contains logical connections, which connect traffic end-nodes and VNF-capable nodes. It implements the chains of VNFs.

NL the Network Layer contains the IP connections, which connect traffic end-nodes, VNF-capable nodes, and networking nodes. It implements the network paths.

sharing among slices can be considered then in NL only or both layers. We assume the latter case and full sharing.

A. Slicing model concept

From the service requirements point of view, in our simplified model concept, each slice defines a traffic type with an ordered set of required VNFs and QoS values. Moreover, this traffic type is described with the high- or low-level traffic parameters, as request arrival rate, interarrival time, and length of packets. For example, let us refer to a slice supporting voice and another supporting real-time video calls.

Dedicating the resources to slices helps to provide guarantees, but can lead to suboptimal usage and lower throughput in several situations when dynamic traffic changes occur. The analysis of this concept is out of our current focus. We assume no dedication in the shared model, i.e., all the slices can use any network resources. A mechanism for coordinating the use of resources in FL and NL is needed to support the requirements. How the VNF resources like CPU/time or memory can be assigned to traffic requests of different slices is out of our scope now, and we assume no limitation in the functional layer.

In the networking layer, we can assume the resource sharing supported by traffic management techniques like in DiffServ or IntServ model of IP. To follow our simple concept of slicing, the class-based approach of DiffServ can be enough for handling slice traffic on IP links. In this case, the link capacity sharing can be implemented with weight-based queueing like Weighted Fair Queueing (WFQ) or its version Low Latency Queueing (LLQ), which also supports strict priority class.

This time we consider only unicast requests. It is worth mentioning that not like in many other works, e.g., in [11], in our concept, the slice is not restricted to only one possible pair of end-nodes. Instead, it supports a set of such relations, i.e., traffic requests of the same slice can have different endpoints. The important is that they are of the same type.

Various technologies can solve the implementation of the two-layer model and the slice traffic management in NL:

- GRE (or other) tunnels can realize FL links,
- IP routing, like OSPF can realize the mapping of FL links to NL links,
- MPLS-TE, or IPv6 can provide traffic classification at the entry nodes and class-based handling on links.

B. Modelling Queueing and Overloads

WFQ and LLQ are weight-based serving policies for queueing, which allows the share of link capacity among the traffic of different classes. The class load, or in our case, the slice traffic, can be adjusted on the links in many ways. Let us refer here to two basic cases.

In traffic engineering, weights are set on each link separately, according to the relation of admitted traffic load coming from the supported classes. On the other hand, in the case where the class preferences are determined preliminary and independent from link loads, the required relation of classes can be *coded* in the weights. For instance, we can say that in general, the traffic of slice S_1 shall get twice as large bandwidth as slice S_2. Then we can set the same weights on every link according to the preliminarily determined values. Our model considers this second approach, and no strict priority class is used now.

One can calculate QoS properties for a queueing system using theoretical models, primarily based on the Markovian approach or its extensions like Markov arrival processes (MAP), or quasi-birth-death processes (QBD). These models are stable when the relative load is less than 1, i.e., there is no overload on the system. However, we have to study also networks where the traffic dynamics can lead even to link overload situations. In such cases, the effective load of slice traffic gets reduced to the part that can pass through the link, because the part over the link capacity gets thrown with high probability.

To catch this behavior, we use a simplified approach instead of the exact stochastic model. The model considers directed traffic loads and link capacities, and we illustrate it in Fig.

2. The figure presents three different load use-cases for a link shared among the *red*, *violet* and *green* slices with WFQ weights of $w_r = 0.2$, $w_v = 0.3$ and $w_g = 0.5$. The dashed line shows the link capacity, and the fourth column illustrates the high load case showing also the parts thrown away from the *red* and *green* slice traffic due to overload.

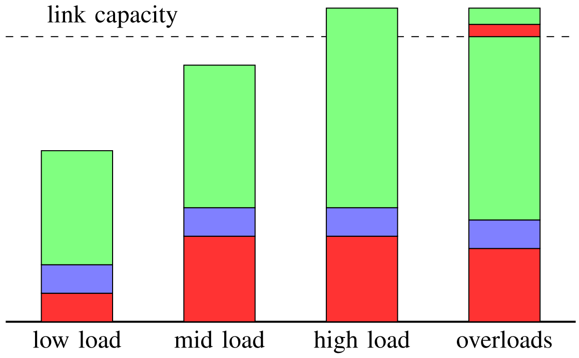


Fig. 2. Link capacity sharing in different load cases

Let B_i be the requested (or offered) bandwidth of slice/class S_i demands on a link, while the average experienced bandwidth for this traffic be $\overline{EB}_i$. Note that on each link l with capacity C_l, we have:

$$C_l \geq \sum_{S_i \in \mathrm{S}} \overline{EB}_i,$$

and,

$$B_i \geq \overline{EB}_i, \forall S_i \in \mathrm{S}.$$

In the simplified model, we assume that in the case of a lower or middle load of a link, i.e., without overloads, the weights do not play a significant role in the level of averages. Thus, we assume:

$$\overline{EB}_i = B_i, \forall S_i$$

For an overloaded link with capacity C_l and using weights w_i in WFQ, we have two cases. Let $\mathrm{S}^{ul} \subset \mathrm{S}$ the subset of slices requesting *less bandwidth than possible*, i.e., where:

$$B_i \leq w_i C_l$$

The model calculates the average bandwidths as follows:

$$\forall S_i \in \mathrm{S}^{ul} : \overline{EB}_i = B_i \tag{1}$$

$$\forall S_i \notin \mathrm{S}^{ul} : \overline{EB}_i = \left(C_l - \sum_{S_u \in \mathrm{S}^{ul}} B_u \right) \frac{w_i}{1 - \sum_{S_u \in \mathrm{S}^{ul}} w_u} \tag{2}$$

The average bandwidth of slices in subset S^{ul} is easily calculated. For the other slices, we start from the capacity remaining for them, and share it according to the WFQ weights normalized on this subset of slices.

The simple model can be extended to consider a strictly prioritized slice S_p too. For S_p, we have:

$$\overline{EB}_p = min(B_p, C_l) \tag{3}$$

The capacity C_l of the link has to be decreased by $\overline{EB}_p$ before the subset S^{ul} gets selected, and the further calculation is performed.

We might use this simple model easily for a single link, but we are in a much more complicated situation with a network of links or queues. As best, we should handle this case by a reduced-load approximation, an iteration on the requested and average bandwidths of slices. However, in this work, we use a simplified approach also for this issue.

C. QoS metrics

The simple model might determine average bandwidth values even for overloaded situations, but a QoS analysis requires a more accurate approach, like that proposed in [20]. It might provide packet-based waiting time and packet loss probability values, and show their dependence on slice weights. We propose a combination of these two levels to get metrics for our analysis.

First, a macroscopic model is involved in handling overloads. According to the simple model above, for each slice, the required and experienced bandwidth will be lost on an overloaded link. Therefore, its interpretation can be a slice load reduction, which comes from overloading. To describe the factor of reduction on a link, we define the value:

$$OvlRed_i = \frac{B_i - \overline{EB}_i}{B_i}$$

A higher factor means more fraction of lost traffic.

To avoid instability, we apply the stochastic model with input parameters mimicking a reduction by $OvlRed_i$ factors, i.e., the analysis can be done for traffic not overloading the link. The simplest way is to enlarge the mean of interarrival time, although higher moments might be affected too. Thus, the waiting times and packet loss results are valid for the part of the traffic that is not thrown due to overloading.

Note that the *macroscopic* loss $OvlRed$ is very important and might be greater than the *microscopic* packet loss by magnitudes if the system is overloaded. Therefore, the macroscopic loss also needs to be considered when comparing the waiting times of different mechanisms or network loads.

From the network point of view, link-level values, i.e., the means and higher moments of the mentioned metrics on single network links might be important. However, for us, more important are the QoS values regarding the traffic requests. Therefore, we extend the proposed metrics starting from the link-level values to end-to-end values as in [21].

We calculate the mean end-to-end latency and packet loss probability metrics for the traffic request r^i coming from slice S_i by applying the simple forms:

$$E^{tr}_{r^i}(W) = \sum_{l \in P_{r^i}} E(W_{l,i}) \tag{4}$$

$$p^{tr}_{r^i} = 1 - \prod_{l \in P_{r^i}} (1 - p_{l,i}) \tag{5}$$

Set P_{r^i} is the set of network links used in the whole service chain assigned to the request.

For the $OvlRed$ metric, we use a different approach than for the microscopic loss because the reduction on one link l of the path P_{r^i} strongly impacts the traffic that arrives at the following link. Therefore, the reduction values can not be considered as independent ones. To model this, we take the maximum value on the path, i.e., we calculate:

$$OvlRed_{r^i} = \max_{l \in P_{r^i}} (OvlRed_{l,i}) \tag{6}$$

For the sake of simplicity, we neglect the accurate reduced-load approximation for both microscopic and macroscopic metrics. Since for each slice requests the service chains need to meet an ordered series of VNFs, the path in the network layer NL can even contain loops, and it is hard to consider them in the iteration.

IV. Slicing-aware Service Chaining

A. Service Chaining Algorithms

Most of the algorithms proposed for service chaining handle traffic requests independently, considering only the required resources like bandwidth or VNFs. Some of them might concentrate only on the number of used networking resources as SFC-CSP [18] does, which ignores the overloading effect. Other algorithms like OdAASP in [17] try to avoid the overloaded network resources if it is possible, but still not consider that the sharing of a resource is usually done on a class or slice basis. It can lead to inefficient use of the links and higher traffic loss when the load increases.

In [19] we present solutions that involve the concept of slicing without systematic introduction of whole class of the algorithms. Their main advantage is the resource preservation for the future requests of a slice. SLF and SLN algorithms use the limitation of resource usage and can control the loads coming from the slices on the Functional or Networking Layer links, respectively. The applied mechanism is simple: the link cost gets vastly increased before the path selection if a special limit rate SL for the given slice would be overridden on the link by leading the chain or the network path of the current request through it.

Both algorithms are based on the lowest cost chain solution SFC-SP [18]. That finds the lowest cost chain from the source s to the destination d of the service request r_i of slice S_i and bandwidth bw_{r_i} considering the series of VNFs prescribed for r_i.

In *SLF*, first of all, the cost $c(l^F)$ of the functional link l^F gets modified to a relatively high value like 10^6 times greater than its original cost, if:

$$bw_{r_i} + \sum_{r \in \mathrm{R}(S_i, l^F)} bw_r > SL(S_i, l^F) \times B(l^F) \tag{7}$$

$\mathrm{R}(S_i, l^F)$ is the set of the already chained requests that are from slice S_i and contain link l^F in their chains. In our two-layer network model, the capacity $B(l^F)$ is calculated as the bottleneck capacity on the network link path to that l^F is mapped.

After modifying costs, the lowest cost chaining finds a chain excluding the links where the slice traffic would be over the limit. Note that the Slice Limitation concept could be applied with any other service chaining algorithm.

The algorithm *SLN* works very similarly to *SLF*, but the relative load limitation is taken into account on the network links of NL. Since more than one functional link can be mapped on a network link, their load cannot be considered independent. The QoS-based prioritization is done on the Network Layer's resources; thus, from *SLN* we can expect service chaining that is more adjusted to packet serving.

The load limitation with *SLN* is based on the values $SL(S, l^N)$ set for each slice S and network link l^N. The algorithm *SLN* starts with the modification of the cost $c(l^F)$ of each functional link l^F, where the mapping of l^F contains at least one network link l^N with

$$bw_{r_i} + \sum_{r \in \mathrm{R}(S_i, l^N)} bw_r > SL(S_i, l^N) \times B(l^N) \tag{8}$$

The functional links that violate the limitation get a high cost, leading to the use of network links, which are not so much loaded by the slice S_i.

On all links, we set the limit value for each slice. The results show that these solutions allow resource sharing among slices according to a preference system defining the priorities or weights of slices. Such kind of preference system is realized as the WFQ or LLQ systems on networking resources. On the other hand, in the low and middle range of load, the algorithms behave nicely also from the overloading point of view since they balance the load among the resources.

B. Slice limitation and QoS

The performance of the algorithms SLF and SLN depend strongly on the limiting parameter SL_i. There are two basic cases of setting SL_i for slice S_i:

- in the simple case, we set the same values on each link *uniformly*, e.g., according to the preferences regarding the slices,
- in the generic case we can set *different* values on every link, e.g., according to the estimated slice load on the link.

Although the generic case might perform better when the network behavior is well estimated or the weighting and limiting parameters are often adjusted, dynamic slice request changes can lead to unstable situations in such a case.

To avoid it, we consider the same settings for queueing weights on each link, and we use the simple setting case in this work. However, it remains a question, which value shall be set as limit SL_i. We might precisely adjust the limits to the set of WFQ weights realized in queues and reflect the provider preferences on the slices, but this is not the only way. We aim to provide and compare different setting policies for the slice-aware algorithms.

C. Adjusting policies

We propose three policies for adjusting the limitation parameters. For a clearer view, we assume normalized w_i weights in WFQ, i.e., $\sum_{S_i \in \mathrm{S}} w_i = 1$. The limit parameters can be set as it follows:

Conservative (Cons) The value:

$$SL_i = \min_{j \in \mathrm{S}} w_j$$

on each link, i.e., the bandwidth of each slice is intended to be pushed below the load of the least weighted traffic class. It means that SFC starts to use low loaded links quite early. This policy is supposed to work nicely only for the case when the weights are similar.

Weight-aware (WeiAw) The value:

$$SL_i = w_i$$

on each link, i.e., the links shall be loaded with slice traffics according to their weights. This policy should preserve as much bandwidth resources for a slice, as it would take with the queueing.

Liberal (Lib) The value:

$$SL_i = \max_{j \in \mathrm{S}} w_j$$

on each link, i.e., the limit for a slice can be higher than its weights. It means that SFC starts to use low loaded links quite late. This should work well in cases where the traffic request pattern is not aligning well with the queueing weights.

Enabling service chains with maybe more hops but less loaded links shall affect the QoS metrics. On the one hand, the higher number of hops in routes can enlarge the end-to-end delays and lead to more queues where the traffic might suffer packet losses. However, on the other hand, we can expect lower values in link-level results for delays and losses in the case of moderate network load.

When the network load elevates strongly, any of these simple policies can lead to situations with high costs on many functional or network links. Thus, the SFC tends to use the shortest chain for many traffic requests, and link overloads can appear.

V. EVALUATION OF THE PROPOSED METHODS

We compare the above introduced QoS results for the policies in topologies of different scales. We have implemented the proposed solutions in the framework presented in [22]. The SFC is performed in this simulation tool considering the order of the requests arrival, but not allowing any departures. Based on the link-loads experienced in the simulator and considering the assumed QoS characteristics of slices, we calculate the metrics applying the theoretical models of [20] and the extensions proposed above.

A. Small topology

First, let us introduce the analysis using the network topology illustrated in Fig. 3, which also shows the VNF capabilities in nodes. Network links are of 1Gbps capacity, and one-to-one mapping is applied between FL and NL. We assume two slices, traffic requests of slice S_0 (red) and S_1 (blue)require VNFs v_0 and v_1 respectively. In the experiment, we evaluate a four-phase elevation of slice traffic, adding more and more requests to slice S_0 between the node pair $E - F$. The traffic of S_1 is less increasing, and its requests are between node pairs $A - C$ and $B - D$. The number of requests to chain, the average of the demanded bandwidth, the mean packet lengths, and the assigned WFQ weights are summarized for each slice in Table I. We apply the WFQ weights on each link uniformly.

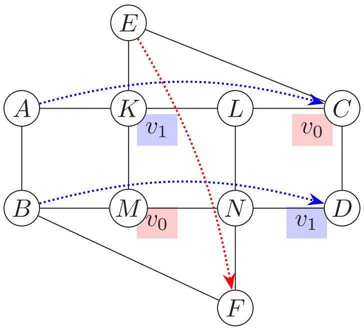


Fig. 3. Toplogy with service chains

TABLE I
TRAFFIC TYPES CHARACTERISTICS

Traffic	Number	Bandwidth	Packet Length	Weight
Slice S_0	1-8	0.3 $Mbps$	8 $Kbit$	0.6
Slice S_1	2-3	0.35 $Mbps$	500 $Kbit$	0.4

Fig. 4 presents on its two y-axes the average values of macroscopic and microscopic loss calculated for the S_1 traffic requests. The policies proposed for the limitation-based algorithm SLN are compared with each other and the simple SFC-CSP. In this case, we applied a one-to-one mapping for links in FL and NL; thus, SLF and SLN are identical.

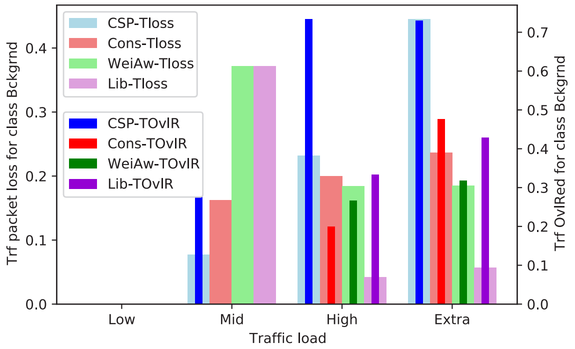


Fig. 4. Average traffic-level loss for slice S_1

As expected, in the case of relatively low and mid-range traffic load, all the metrics are moderate. The elevation of the load induces the elevation of microscopic packet loss, except for the SFC-CSP algorithm, where the overload of links appears already. The best-performing policy here is the conservative one.

In the higher load ranges, we can see that due to the overload of network links, the overload reduction plays the primary role in the loss, while microscopic packet loss decreases. The macroscopic reduction of traffic ends with less loaded links and thus with lower packet loss values. In SFC-CSP, the chains are static, and in SLN with the liberal policy, we accept higher slice loads on links before we start to use alternate chain paths. This behavior leads to higher overloads.

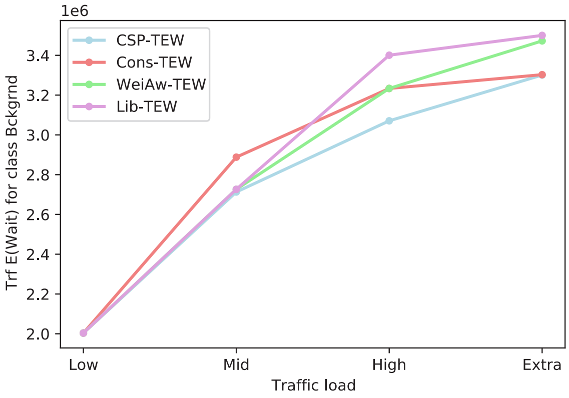


Fig. 5. Average traffic-level waiting time for slice S_1

In the case of SLN with weight-aware, and especially with conservative policy, the traffic suffers from a different effect. The higher load induces a pretty early use of alternate chain paths and balance the load among more links, but these chains might be of more hops. Having more hops in NL means more links where the traffic and the packets can get lost. On the other hand, when the network load gets high, nearly all network resources can get overloaded because traffic flows everywhere. The results show that considering both macroscopic and microscopic loss we get the lowest values for the conservative policy if the load is not too high. For extra high load cases, the best performing policy is the weight-aware, where the SL_i limits are adjusted to the weights used in WFQ.

The above explanations clarify the results on the end-to-end delay of S_1 requests presented in Fig. 5. Although suffers from significantly higher losses, algorithm SFC-CSP, with its relatively short chain paths, performs well from this perspective. The SLN goes better with conservative and weight-aware policies, while the liberal one performs poorly.

B. Larger Topology

The network under the scope is the hypothetical backbone network of Algeria, which is used in [17] too. Fig. 6 presents the topology of the IP layer, which contains 10 Core Routers (black-filled nodes) and 17 Edge Routers (grey-filled nodes) each at different sites.

We assume only one type of IP connection of 10Gbps capacity, attached to Core Routers and Edge Routers. In addition, 27 eNodeBs are distributed on the 27 sites, and three different VNFs are placed by random placement resulting in the following *Core* nodes:

- v_1 is placed in Algiers and Boussada,
- v_2 is placed in Algiers and Tenes,

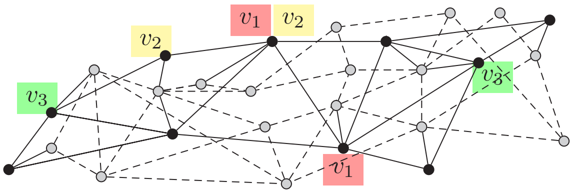


Fig. 6. IP layer topology and VNF placement

- v_3 is placed in Oran and Constantine,

The links of the functional layer FL form a full graph between these three nodes extended by those connecting the eNodeBs to these nodes. The mapping onto network links in NL simply uses the shortest paths.

There are two simplex traffic demand types, in other words, slices, to be served, New Services and Best Effort, referred to as S_0 and S_1, respectively. Each traffic demand requires to pass through the VNF-series v_3, v_2, v_1. The demands of the New Services type start from one eNodeB in Algiers to all eNodeBs. Each traffic demand of type Best Effort starts and ends in randomly chosen eNodeBs. We aim to study a scenario where the type of New Services S_0 bandwidth grows linearly from 0 up to $1500 Mbps$. The traffic demands arrive in a randomized order. The numbers of demands, the average of the demanded bandwidth (in Mbps), packet length, and WFQ weights are summarized for each traffic type in Table. II. In the SL-based algorithms we apply the Weight-Aware policy.

TABLE II
TRAFFIC TYPES CHARACTERISTICS

Traffic	Number	Bandwidth	PacketLen	Weight
New Serv. (S_0)	27	$0 - 1500$	$8\ Kbit$	0.6
Best Effort (S_1)	702	$3,98$	$500\ Kbit$	0.4

Fig. 7 and 8 present the average end-to-end packet delay calculated for slices S_0 and S_1.

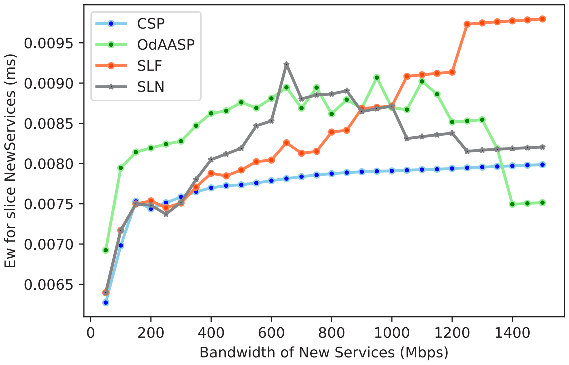


Fig. 7. Average traffic-level delay for slice S_0

In the low-load range, below 450 Mbps, CSP and SL-based algorithms perform nearly the same while values for OdAASP are slightly higher. In the mid-load range, 450-900 Mbps, on the one hand, the SL-based methods show higher waiting time, which comes from finding low-loaded links in the corresponding slice and use them. The network resources get exhausted in the high-load range, over 900 Mbps, where we observe very high waiting times. Here we also face a descending trend in OdAASP and SLN, which comes from returning to choose often SFCs with the links of the shortest network paths.

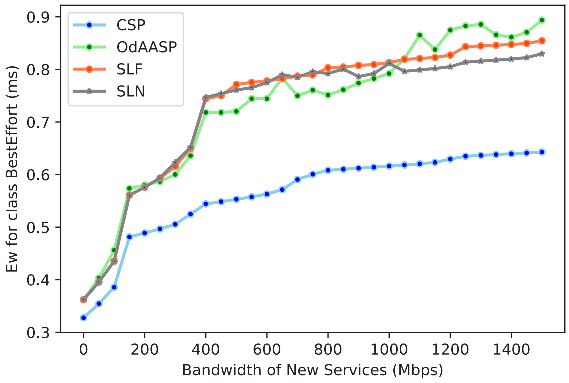


Fig. 8. Average traffic-level delay for slice S_1

We observe similar behavior in Fig. 8 for the other slice, where the SLN algorithms perform in the same way as OdAASP, without the descending trend at high loads. In the case of CSP, the waiting time values are low because chains' paths contain few links in both the Functional and Networking Layer.

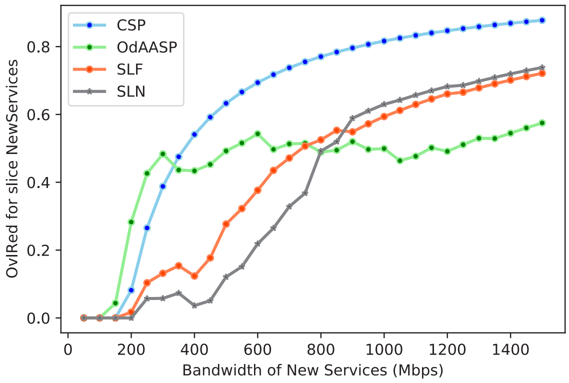


Fig. 9. Average traffic overload reduction for slice S_0

In the CSP case, it is evident that the waiting time is lower than for the others since it always uses the shortest paths when chaining the demands. The other algorithms do this when the load is high, and there are already many demands using the links in FL or NL. However, as expectable, always using the same links in CSP leads to overloads even for moderated network load. In Fig. 9 we observe that the demands of S_0 suffer a pretty high overload reduction with CSP. The overload avoiding algorithm OdAASP performs worse than the SL-based ones in the low or middle load phase due to the high number of demands in slice S_1. Unlike SLF and SLN, OdAASP does not spare resources for S_0 traffic and might chain it over short but overloaded paths.

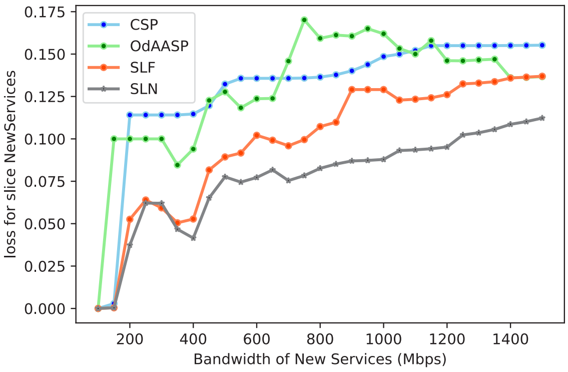


Fig. 10. Average traffic-level loss for slice S_0

Fig. 10 presents the end-to-end packet loss or microscopic loss that happens on links after throwing away the overloading parts. Also, here we observe that the SL-based algorithms perform better than CSP and OdAASP, although the difference is not that large for higher loads. The cause of the fall-back effect at about $300Mbps$ is that link overloads appear in the network. The macroscopic traffic reduction affects the microscopic metrics in a good direction, i.e., lower losses and waiting times.

We observe that SLN with weight-aware policy goes pretty better than any other algorithm. As the relative load limitation is considered on the network links of NL, it takes more chains that can be mapped on a network link. The QoS-based prioritization is done on the Network Layer's resources; thus, from SLN, we can expect service chaining that is more adjusted to packet serving.

VI. Conclusion and further work

This paper focuses on SFC methods, which support slices, and consider their packet level handling during the calculation of the appropriate service chain. It proposes different policies for setting up the parameters of the SFC methods. The model behind the methods ignores the load and latency details or limitations of VNFs, but considers link capacities and network loads coming from the different slices, which share the available resources according to the implemented queueing. This allows the systematic evaluation of QoS properties that can be experienced on the links or by the service requests. Result values are calculated with a packet-level model of network links extended by a macroscopic loss concept that shall handle overloads. The concept is not strictly coupled to the NFV, and might be applicable for architectures based on containerized functions.

The calculation model could be extended to analyze algorithms' performance for slices supporting time-critical applications. Besides the average latency, we could calculate their maximum values or at least the probability of overriding a given delay threshold.

The numerical results show that the proposed algorithms perform better from several QoS metrics point of view than those missing the slice-aware property. From a set of results seems that the conservative policy works well when the load is moderate, while for higher loads the weight-aware policy can perform better. Another result-set demonstrates the advantages of method SLN in the middle load range, although the evaluation is not straightforward due to the complex dependencies among the different kinds of measures.

Moreover, handling the requests dynamically while optimizing the SFC separately for the slices may lead to handle the slices in an unfair manner. As a next step, the further analysis from fairness point of view shall be done. The extension of the proposed policies might help to catch this issue.

References

[1] ETSI. Network functions virtualisation (nfv)management and orchestration. [Online]. Available: https://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_NFV-MAN001v010101p.pdf (Accessed 2014-12)).

[2] B. Han, V. Gopalakrishnan, L. Ji, and S. Lee, "Network function virtualization: Challenges and opportunities for innovations," *IEEE Communications Magazine*, vol. 53, no. 2, pp. 90–97, Feb 2015.

[3] R. Mijumbi, J. Serrat, J. Gorricho, N. Bouten, F. De Turck, and R. Boutaba, "Network function virtualization: State-of-the-art and research challenges," *IEEE Communications Surveys Tutorials*, vol. 18, no. 1, pp. 236–262, Firstquarter 2016.

[4] J. Halpern and C. Pignataro. (2020) Service function chaining (sfc) architecture. [Online]. Available: https://datatracker.ietf.org/doc/rfc7665/?include_text=1

[5] G. P. Sharma, W. Tavernier, D. Colle, and M. Pickavet, "Vnf-aap: Accelerator-aware virtual network function placement," in *2019 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*, 2019, pp. 1–4. doi: 10.1109/NFV-SDN47374.2019.9040061

[6] J. Pei, P. Hong, M. Pan, J. Liu, and J. Zhou,"Optimalvnfplacementvia deep reinforcement learning in sdn/nfv-enabled networks," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 263–278, 2020.

[7] N. alliance. 2016 ngmn 5g p1 requirements and architecture work stream end-to-end architecture description of network slicing concept. [Online]. Available: https://www.ngmn.org/wp-content/uploads/160113_NGMN_Network_Slicing_v1_0.pdf (Accessed 2014-07-15).

[8] T. Truong-Huu, P. Murali Mohan, and M. Gurusamy,"Servicechainembedding for diversified 5g slices with virtual network function sharing," *IEEE Communications Letters*, vol. 23, no. 5, pp. 826–829, May 2019.

[9] R. Addad, M. Bagaa, T. Taleb, D. L. Cadette Dutra, and H. Flinck, "Optimization model for cross-domain network slices in 5g networks," *IEEE Transactions on Mobile Computing*, pp. 1–1, 2019.

[10] R. A. Addad, M. Bagaa, T. Taleb, D. L. C. Dutra, and H. Flinck, "Optimization model for cross-domain network slices in 5g networks," *IEEE Transactions on Mobile Computing*, vol. 19, no. 5, pp. 1156–1169, 2020.

[11] A. Chiha, M. V. D. Wee, D. Colle, and S. Verbrugge, "Network slicing cost allocation model," *Journal of Network and Systems Management*, vol. 28, no. 3, p. 627–659, 2020.

[12] S. Vittal, M. K. Singh, and A. Antony Franklin, "Adaptive network slic- ing with multi-site deployment in 5g core networks," in *2020 6th IEEE Conference on Network Softwarization (NetSoft)*, 2020, pp. 227–231. doi: 10.1109/Net-Soft48620.2020.9165512

[13] M. O. Ojijo and O. E. Falowo, "A survey on slice admission control strategies and optimization schemes in 5g network," *IEEE Access*, vol. 8, pp. 14 977–14 990, 2020.

[14] Q.-T. Luu, S. Kerboeuf, and M. Kieffer, "Uncertainty-aware resource provisioning for network slicing," *IEEE Transactions on Network and Service Management,* vol. 18, no. 1, pp. 79–93, 2021.

[15] W.-K. Chen, Y.-F. Liu, A. De Domenico, and Z.-Q. Luo, "Network slicing for service-oriented networks with flexible routing and guaranteed e2e latency," in *2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, 2020, pp. 1–5. doi: 10.1109/SPAWC48557.2020.9154330

[16] R. Gour, G. Ishigaki, J. Kong, and J. P. Jue, "Availability-guaranteed slice composition for service function chains in 5g transport networks," *IEEE/OSA Journal of Optical Communications and Networking*, vol. 13, no. 3, pp. 14–24, 2021..

[17] K. Mebarkia and Z. Zsoka, "Service traffic engineering: Avoiding link overloads in service chains," *Journal of Communications and Networks*, vol. 21, no. 1, pp. 69–80, Feb 2019.

[18] G. Sallam, G. R. Gupta, B. Li, and B. Ji, "Shortest path and maximum flow problems under service function chaining constraints," in *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications,* 2018, pp. 2132–2140. doi: 10.1109/INFOCOM.2018.8485996

[19] Z. Zsoka and K. Mebarkia, "Slice-aware service chaining," in *2021 24th Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, 2021, pp. 6–12. doi: 10.1109/ICIN51074.2021.9385550

[20] A. Horvath, G. Horvath, and M. Telek, "A joint moments based analysis of networks of map/map/1 queues," *2008 Fifth International Conference on Quantitative Evaluation of Systems*, pp. 759–778, 2008.

[21] K. Mebarkia and Z. Zsoka, "Qos modeling and analysis in 5g backhaul networks," in *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Sep. 2018, pp. 1–6. doi: 10.1109/PIMRC.2018.8580739

[22] B. Farkas and Z. Zsoka, "Augmenting sdn by a multi-layer network model," in *2016 European Conference on Networks and Communications (EuCNC)*, 2016, pp. 215–219. doi: 10.1109/EuCNC.2016.7561035

Khalil Mebarkia received the M.Sc. degree in Computer Science (Networks and Multimedia) from University of Bordj Bou Arreridj, Algeria in 2016. He is currently working toward the Ph.D. degree in Computer Engineering at Department of Networked Systems and Services in Faculty of Electrical Engineering and Informatics of Budapest University of Technology and Economics, Hungary. His research interests include Computer Networks and Protocols, Software Defined Networks, Virtualized Network Function, and Cloud Computing.

Zoltán Zsóka received his M.Sc. degree in 1999 in Technical Informatics from Technical University Budapest, Hungary and his Ph.D. degree in 2007 from the same University. He is currently associate professor in Department of Networked Systems and Services, in Faculty of Electrical Engineering and Informatics of Budapest University of Technology and Economics. His research interest includes Networking Virtualization and Automatization, and Software Defined Networks.

A Novel UWB Monopole Antenna with Reconfigurable Band Notch Characteristics Based on PIN Diodes

Yahieal Alnaiemy, and Lajos Nagy

***Abstract*—Our design for a novel UWB monopole antenna structure with reconfigurable band notch characteristics based on PIN diodes is presented in this paper. The proposed antenna is comprised of a modified circular patch and a partial ground plane. The band-notch characteristics are achieved by etching a slot on the partial ground plane and inserting three PIN diodes into the slots for adjusting the operating antenna bands. The reconfigurability is achieved by adding three PIN diodes to obtain eight states with UWB, dual and triple operating bands which can be obtained by changing the PIN state from ON to OFF, and vice versa. The proposed design shows a simple biasing process to switch the frequency bands with insignificant gain variation and low radiation efficiency reduction. The reconfigurability of the frequency is accomplished by adjusting the effective slot length through modifying the PIN diodes states at the desired operating bands. The desired operating frequency bands can be obtained by switching the diodes. A systematic parametric study based on a numerical analysis is invoked to verify and refine the proposed performance. The proposed antenna is fabricated on FR-4 substrate with dimensions of $50 \times 60 \times 1\ mm^3$. The proposed antenna performance was tested experimentally and compared to the simulated results from CSTMW based on FIT. Experimental results were in concordance with simulated results. We found that the proposed antenna design had simple geometry and it was easy to control the frequency bands to suit the applications of WiMAX and WiFi systems.**

***Index Terms*—UWB, Notch band, Reconfigurable antenna, PIN diode, Monopole antenna.**

I. Introduction

DUE to the rapid growth in wireless communication technology, multi-band antennas are highly valued. A significant drawback of conventional multi-band antennas is that they work only for specific operations. A new antenna with desired frequency specifications is required for each new application, increasing the relative cost. Reconfigurable antennas are usually based on active electronic devices to independently tune the required frequency, radiation, and polarization. They have come into urgent demand due to the growth in mobile communication technologies[1]. Reconfigurable antennas have several advantages over conventional antennas such as smaller size, steerable radiation patterns, selecting the desired polarization for different frequency bands, and polarization, which reduces antenna system size and intersymbol interference (ISI) impacts [2]. The integration of these configurations into a single antenna is a major challenge that researchers have been faced in recent years. Integrating antennas with modern high-speed semiconductors such as Positive Intrinisic Negative (PIN) diodes, Radio Frequency Microelectromechanical Systems (RF MEMS), and varactor diodes have been designed successfully for frequency, polarization, and pattern tunability [3]-[25]. Due to the high-speed response and low forward resistance of PIN diodes, it was common practice to vary the antenna performance [3] for direct antenna modulation (DAM) including Differential Phase Shift Keying (DPSK) modulation [4], which makes it more competitive for cognitive radio applications. A slotted patch antenna with two PIN diodes was presented by Majid *et al* for frequency and pattern reconfiguration [5]. For antenna beam reconfiguration, a flexible antenna based on eight PIN diodes for wireless applications was introduced by Zhu *et al* in [6]. For frequency and pattern reconfiguration, an antenna based on two PIN diodes positioned within the slot etched in the front antenna patch was presented by Han *et al* in [7] for Long-Term Evolution (LTE). A reconfiguration of the antenna frequency and pattern was achieved using a printed antenna based on three Radio Frequency (RF) switches [8]. A reconfigurable Ultra-Wide-Band (UWB) filter antenna was featured in [9]; controlling operating frequency bands based on three PIN diodes that directly control the desired WLAN and WiMAX frequency bands. Wu *et al* in [10] introduced and designed a reconfigurable quad-band antenna based on Micro-Electomechanical-Systems (MEMS) switches. Controlling operating frequency can be obtained by adjusting the MEMS switch, making it suitable for the cognitive radio base station. Hamid *et al* in [11] discussed and modelled a Vivaldi antenna by inserting four switchable ring slots within the antenna ground plane structure. The antenna reconfiguration may work in broadband or narrowband mode by changing the PIN diode switching facility. For single, multi-band, and UWB spectrum, a frequency reconfigurable microstrip antenna was presented by Yadav *et al* in [12], based on an array of 27 PIN diodes integrated into the partial ground structure. Yadav *et al* used many RF switches, increasing cost, manufacturing, and measurement difficulties relative to our antenna design. Tasouji *et al* in [13] presented a printed UWB slot antenna based on two PIN diodes with reconfigurable band-notch features that was mounted across the circular slot antenna patch to produce single and double band-notch characteristics. A UWB

Yahiea Alnaiemy is with Budapest University of Technology and Economics, Budapest, Hungary, and University of Diyala, College of Science e-mail: (yahiea@hvt.bme.hu).

Lajos Nagy is with the Budapest University of Technology and Economics, Department of Broadband Infocommunications and Electromagnetic Theory, Budapest, Hungary e-mail: (nagy.lajos@vik.bme.hu).

Manuscript received July1, 2021; revised July 30, 2021.

DOI: 10.36244/ICJ.2021.3.4

monopole antenna with a reconfigurable band-notch based on two PIN diodes placed in antenna patch slots was presented by Han *et al* in [14]. A reconfigurable microstrip slot antenna was presented by Oraizi *et al* in [15] through regulation the PIN diode embedded in the rectangular Split Ring Resonator (SRR). A reconfigurable UWB circular wide-slot antenna based on a stepped impedance resonator and an arc-shaped parasitic element was presented by Li *et al* in [16]. A notched-band UWB monopole antenna was introduced by Aghdam *et al* in [17] by connecting a varactor diode to a π-shaped patch. A reconfigurable cavity-backed slot antenna substrate integrated waveguide based on MEMS active elements was presented by Saghati *et al* in [18]. A dual-band reconfigurable antenna based on a varactor diode lumped into the slot antenna was introduced by Behdad *et al* in [19]. A reconfigurable circular monopole based on Field Effect Transistor (FET) was presented in by Aboufoul *et al* in [20] for cognitive radio applications. A reconfigurable frequency band monopole with single and dual bands by employing three PIN diodes was presented by Shah *et al* in [21]. Nikolaou *et al* in [22] controlled the frequency resonance by switching two PIN diodes soldered on both sides of an annular slot patch antenna. Kim *et al* in [23] were proposed the polarization reconfigurability of a single feed circular patch antenna with five PIN diodes for low frequency and high frequency applications. Elwi in [24] presented a reconfigurable antenna with remotely controlled by integrating a photo resistor array into a Hilbert patch antenna and adjusting the photo-resistors elimination for modern 5G applications. Singh *et al* in [25] were proposed a reconfigurable antenna for tuning the frequency bands in internet of things (IoT) systems using three PIN diodes. The researchers for wireless applications have presented several wideband monopole antennas. For instance, Reddy *et al* in [26] designed a flexible wideband monopole antenna for body-centric wireless communications. While, Mohandoss *et al* in [27] designed a planar monopole fractal antenna to enhance the bandwidth for the personal wireless area and UWB applications. A multiband reconfigurable microwave filtering monopole antenna was presented by Kingsly *et al* in [28] based on switchable agile multiband filtenna for cognitive radios and Time Division Multiple Access (TDMA) systems. Finally, compact wideband flexible planar monopole antennas were designed and analyzed for body-centric wireless and UWB communications [29].

The main objective of using such reconfigurable UWB frequency antenna is to overcome the overlap of the UWB with other bands, such as WiMAX bands. The antenna performances are obtained from a comparative study of the conventional antenna with the proposed antenna based on PIN diodes within the ground plane. This paper introduces a well-controlled operating frequency band based on three $ON-OFF$ switch statues. By changing the slot length effectively through the PIN diode switching, a frequency band reconfiguration can be achieved at the desired operating bands accordingly. The proposed antenna shows a frequency band from 1.86 GHz to 10.89 GHz that is significantly affected by PIN diodes switching. Eight cases, therefor, can be generated from switching the proposed three PIN diodes to provide UWB, two, and/or three operating bands. The simulated results are compared to the experimental results, which show acceptable agreement and confirm good performance of the proposed antenna. The obtained results, therefore, confirm that the proposed reconfigurable antenna is a better candidate for integration into wireless communication circuits.

II. Antenna Design Detials

The proposed antenna is designed to operate from 1.89 to 10.89 GHz to achieve a UWB with matching impedance in terms of S_{11} bellow -10dB according to the IEEE standard [15]. To be suitable for near and medium communication distance, the antenna's peak realized gain should be within the range of 5.2dB-6.03dB for WiFi applications. The proposed antenna design, with the suggested parameters, is depicted in the next section. The antenna is modeled step by step to obtain a UWB response with an impedance bandwidth of (S_{11} < -10 dB) for the entire UWB (1.85-10.9)GHz. Next, three PIN diodes are connected to the ground plane slot as switches to obtain eight PIN diode states. The main reason to choose three PIN diodes as switches is to change the proposed antenna's effective electrical length to achieve frequency reconfigurability. By switching the state of the PIN diode between forward and backward, we can achieve the proposed antenna design to operate in UWB, dual, and triple-band mode.

III. Methodology of the Antenna Design

In this section, we present the geometrical details of the proposed antenna to provide the design methodology for the optimal antenna performance. Next, antenna reconfiguration is discussed by switching the PIN diodes. The reconfigurability of the antenna can be accomplished by employing three PIN diodes as switches in the simulation setting to obtain dual-band mode, triple-band mode and UWB depending on the switch state. The authors have chosen a proposed monopole antenna, based on the circular shape and a defected ground structure that could expand the frequency band of operating as can be seen later in this paper.

A. UWB Monopole Antenna Design

The configuration of the proposed UWB monopole antenna geometry with the design parameters is illustrated in Fig. 1 and Table I, respectively. The proposed antenna is fabricated on a low-cost FR4 substrate with a ϵ_r of 4.3, and a tanδ of 0.025. The proposed antenna size is $50mm \times 60mm$ with a substrate thickness of 1 mm. The modified circular patch is printed on the top side, and the defected partial ground plane is printed on the bottom side. The radiating patch and ground plane shapes are modified to achieve a UWB with good impedance matching.

In this section, reach the final antenna design is discussed. The antenna design methodology was proposed by using a commercial software package of Computer Simulation Technology Microwave Studio (CSTMW) with Finite Integration Technique (FIT) algorithm [30]. For this, the antenna bandwidth is monitored to get the best matching impedance over

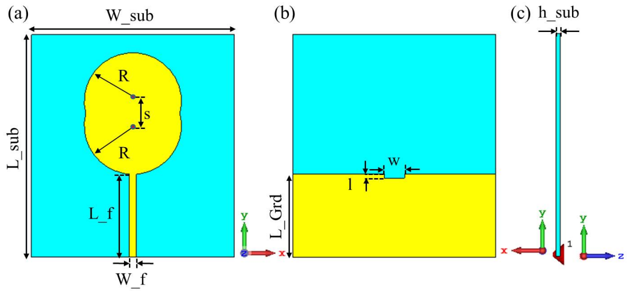


Fig. 1. The proposed antenna geometry; (a) front view, (b) back view and (c) side view.

TABLE I
The Geometrical dimensions of the proposed antenna: all dimensions in mm.

Parameter	Dimension	Parameter	Dimension
L_sub	60	L_f	23
W_sub	50	W_f	1.65
L_Grd	23	R	12
h_sub	1	S	6
w	5	l	1

the entire band of interest by feeding the antenna with a 50 Ω port. To determine the influence of changing antenna dimensions, a parametric study was conducted. Antenna performance was monitored with respect to changing W_f, W, l, S, h_sub, and L_Grd parameters representing feed line width, slot width of the partial ground plane, slot height of the partial ground plane, center-to-center distance, substrate height, and ground plane length, respectively. Next, a parametric study was applied to find the appropriate substrate type and size. The antenna design begins with a conventional circular patch backed by a full ground plane as mentioned in case 1 (Fig.2). Next, the ground plane is changed to a partial ground plane (without any slots) to improve the impedance matching, as mentioned in case 2. Next, the rectangular slot is etched from the ground plane to improve the proposed antenna bandwidth. The frequency reconfiguration is accomplished by modifying the parasitic element electrically through switching the PIN diodes with the ground plane. The defected ground plane is made by integrating the PIN diodes without changing the UWB antenna performance. We observed that the desired UWB results are not obtained for the proposed antenna with partial ground plane, therefore, a slot is inserted within the partial ground plane to improve the bandwidth and the matching impedance. We observed that the antenna bandwidth of 3.5 GHz to 9.5 GHz was obtained as mentioned in case 3. A further modification of the proposed antenna is to involve a circular patch as denoted in case 4 of Fig. 2. The modification in this design includes another circular shape to generate a new frequency band. This modification in the radiator patch obtains a better impedance matching over the entire UWB and enhances the reference antenna bandwidth as shown in case 4. We observed a very wide bandwidth (S_{11} < -10 dB) of 1.85 GHz to 11 GHz.

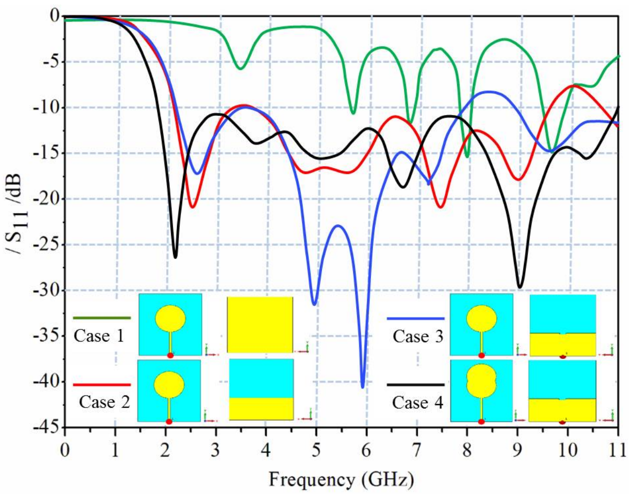


Fig. 2. Comparison of simulated $|S_{11}|$ spectra for four model cases.

It is indicated from the S_{11} spectra Fig. 2 that variation in the ground plane has significant effects on the antenna bandwidth. For example, the proposed antenna with full ground plane in case 1 shows narrow bandwidth around multiple frequency bands. Antenna bandwidth, therefore, is enhanced when a partial ground plane is modified. However, the antenna bandwidth is enhanced furthermore in cases 3 and 4 when the slot is introduced to the ground plane. The results are in agreement with the suggested antenna design's requirements which are provided in Section II for achieving a wide-band with matching impedance below -10 dB.

To maximize the antenna bandwidth further, the antenna feed line was connected to the Sub Miniature version A connector (SMA) center pin with a width of W_f. The feed line width is parametrically optimized as shown in Fig.3. It is indicated that varying W_f shows a major impact on the antenna bandwidth [24]. This study may lead to the best impedance matching and subsequently, a gradual bandwidth increase in the proposed antenna. Figure 3 shows the optimized results of W_f at 1.65 mm with broader bandwidth over the frequencies from 1.85 GHz to 10.89 GHz. This parametric study considers S_{11} < -10 dB impedance bandwidth for the entire UWB by changing W_f given the antenna wide band from 1.85 to 10.85 GHz.

The main purpose of the rectangular slot on the ground plane is to disturb the surface current to operate the antenna at lower frequency bands. The effects of changing the slot width (W) on the $|S_{11}|$ spectra (Fig. 4) was studied to realize the best matching impedance. We found that decreasing the slot width (W< 5 mm) causes impedance matching reduction from 2.5 GHz to 7.8 GHz in the low-frequency band, while an increase in slot width (W> 5 mm) creates problems with impedance matching from 7.2 GHz to 7.8 GHz in the high-frequency band. The maximum impedance bandwidth and best impedance matching can be obtained when W = 5 mm. A sweep of the parameter W to achieve the proposed antenna work with a UWB mode with impedance bandwidth less than -10 dB from 1.85 GHz to 10.9 GHz.

To maximize the antenna bandwidth, the height of the slot

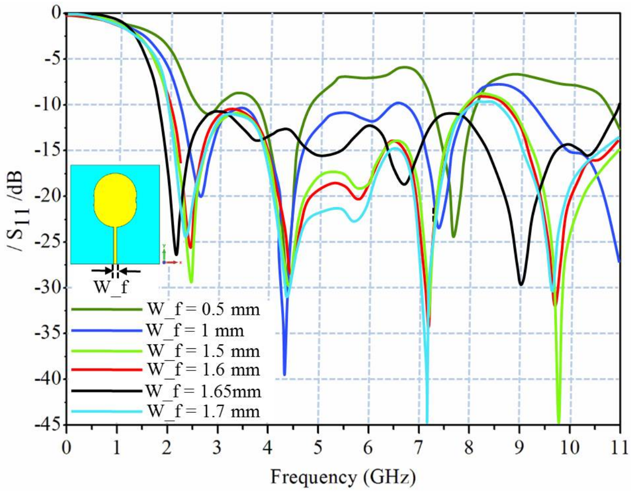


Fig. 3. Effects of changing (W_f) on the $|S_{11}|$ spectra

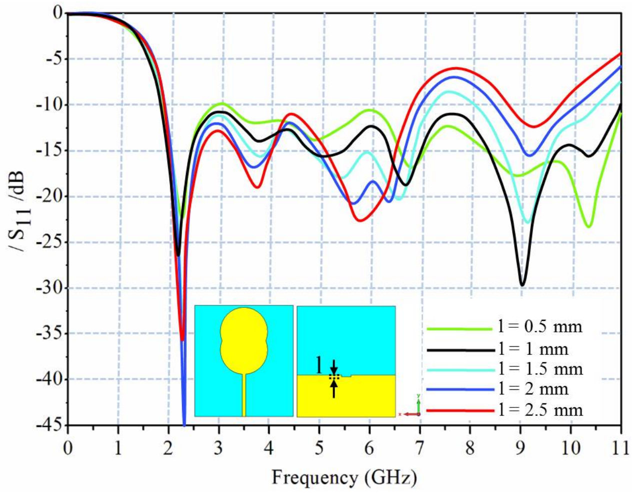


Fig. 5. Effects of changing l on the $|S_{11}|$ spectra

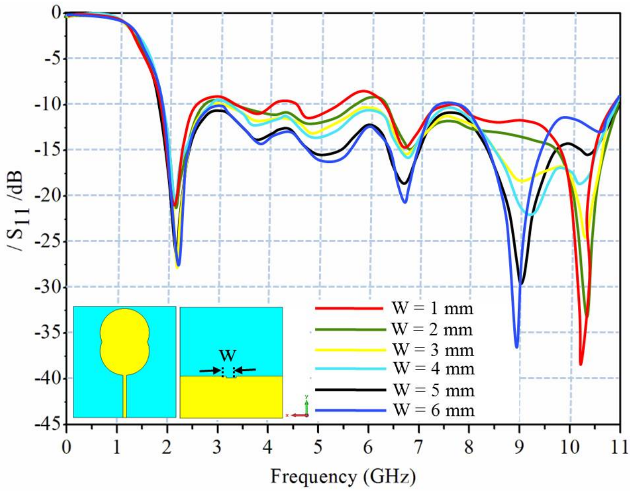


Fig. 4. Effects of changing W on the $|S_{11}|$ spectra

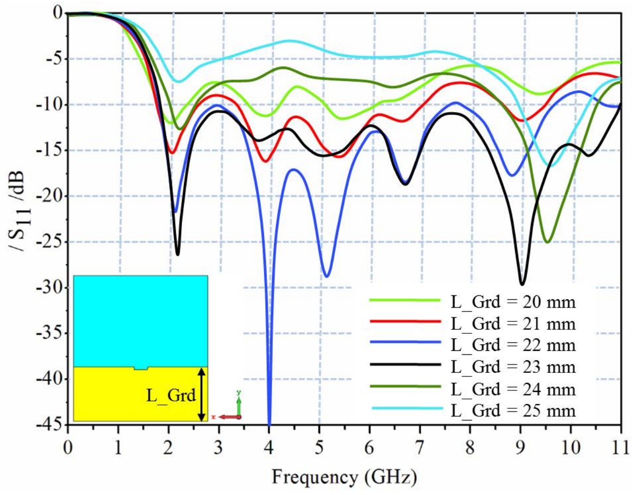


Fig. 6. Effects of changing L_Grd on the $|S_{11}|$ spectra

on the partial ground plane has a pivotal function. However, We studied the effect of the slot height of the partial ground plane given by (l) on the proposed antenna performances. The effect on bandwidth for different ground plane slot heights is shown in Fig. 5.

As shown in Fig. 5, the narrower slot height (l= 0.5 mm) indicates poor return loss at the 2.9 GHz to 3.2 GHz band. Increasing the slot height (l > 1 mm) leads to enhanced $|S_{11}|$ at the lower band (1.75-6.75) GHz, but poor impedance matching is found between 6.75 GHz and 8.5 GHz. However, the optimum ground slot height (l= 1 mm) provides the necessary impedance matching over the required frequency range, and the UWB width of the antenna is from 1.75 to 11 GHz. These results were extremely close to the planned antenna specification, as shown in Section II. The length of partial ground plane (L_Grd) shows a slight effect on the proposed antenna bandwidth. The L_Grd was swept from 20 mm up to 25 mm with a step of 1 mm to obtain the desired UWB frequency band. We observed from Fig. 6 that the proposed antenna operates as a UWB antenna when L_Grd = 23 mm. However, $|S_{11}|$ is improved dramatically when the ground plane length is gradually reduced. As a result, the required antenna design criteria are met by selecting L_Grd = 23 mm for the entire UWB mode with the best bandwidth impedance matching less than -10 dB.

Figure 7 shows the circular patch development through five stages to illustrate the effects of changing S value from 0 mm to 8 mm with a step of 2 mm. The parameter S represents a center-to-center distance between two circular patches (Fig.1). In this design step, the simulated $|S_{11}|$ spectra of the proposed antenna with a different value of S is illustrated in Fig. 7.

At a low value of S (S< 6 mm), the antenna does not provide a UWB impedance bandwidth. The proposed antenna shows poor matching impedance in the high- frequency band from 7.2 GHz to 8.1 GHz. Increasing S above 6 mm, improves the impedance bandwidth. However, the antenna still suffers from a matching impedance problem at the frequencies from 7.3 GHz to 8.3 GHz. The distribution of the surface

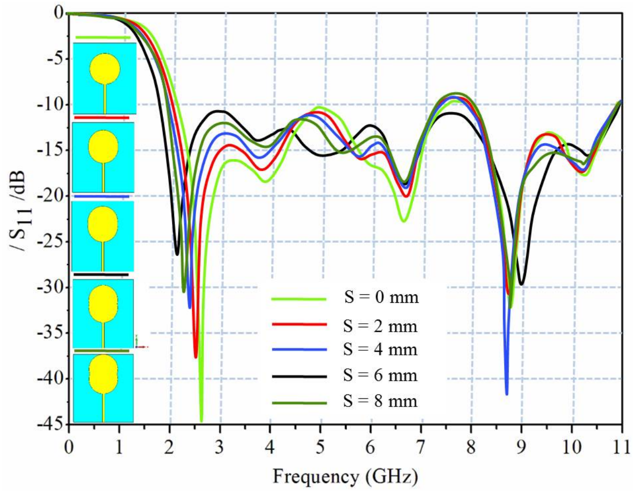


Fig. 7. Simulated $|S_{11}|$ spectra for the proposed patches.

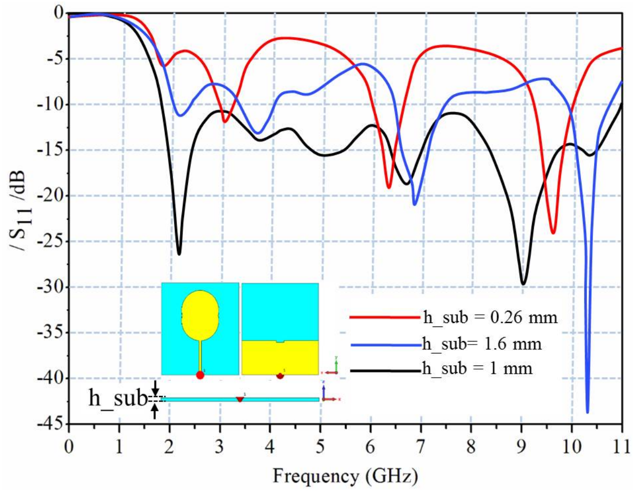


Fig. 8. Effects of changing h_sub on the $|S_{11}|$ spectra.

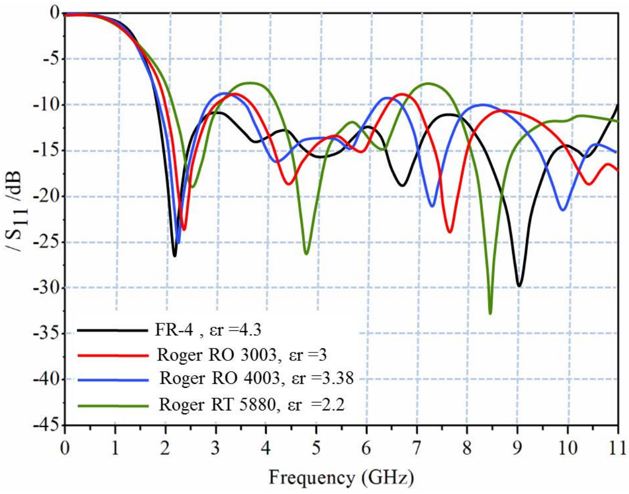


Fig. 9. Effects of changing substrate materials on the $|S_{11}|$ spectra.

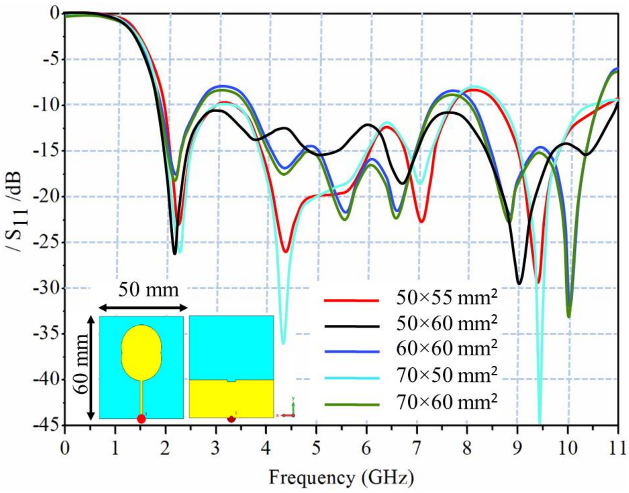


Fig. 10. Effects of substrate size on the $|S_{11}|$ spectra.

current of the proposed antenna with S = 6 mm is mainly extended. By considering S = 6 mm, therefore, the impedance matching bandwidth is significantly enhanced to suit the proposed antenna requirement to work with UWB mode.

Figure 8 demonstrates the effects of changing the substrate thickness on the $|S_{11}|$ spectra. The substrate thickness was changed according to the available commercial resources :0.26, 1, and 1.6 mm. Based on the $|S_{11}|$ spectra, the proposed antenna shows a wider impedance bandwidth for h_sub = 1.6 mm but cannot support the desired UWB. The proposed antenna depicts a wide impedance bandwidth only when it has substrate thickness of 1 mm. As a result, if h_sub = 1 mm , the proposed antenna operates in UWB mode with a matching impedance of less than -10 dB from 1.85 to 10.9 GHz.

Additionally, we discuss the effects of substrate type change on antenna performance. As shown in Fig. 9, FR-4 substrate with ϵ_r= 4.3 and $tan\delta$ =0.025 covers a wider bandwidth compared to other substrates in this study, which satisfied the entire UWB.

Regarding substrate size, we adjusted the proposed antenna dimensions within five different dimensions. As shown in Fig. 10, we found the best antenna dimensions are 50 mm×60 mm, which satisfied the entire UWB.

B. Reconfigurable UWB Monopole Antenna

The reconfiguration of the proposed antenna, which has a slot on the ground plane, is shown in Fig.11(a). The slot on the ground plane is mounted by a narrow a metal strip with dimension 0.5 mm×4 mm to provide independent DC biasing for PIN diodes as shown Fig.11(a). The frequency configuration is changed by adjusting three PIN diodes connected to the ground plane slot. In the RF domain, PIN diodes are commonly used, therefore, the PIN diode RF resistance is connected to the DC bias current and can be used as an RF switch. When the diode is positively biased, the short circuit is turned on, and the open circuit is turned ON when it is reverse biased. Usually, the diode resistance can vary from 10 $k\Omega$ to less than 1Ω by controlling its bias current [32]. In UWB mode, switching the PIN diodes may dramatically affect the impedance matching that reduces the bandwidth due to parasitic inductance, capacitance, and resistance. Nevertheless, the DC biasing circuit is necessary to control the RF signal flow to the antenna structure. For this, integrating the PIN diodes into the ground plane has minimum effect compared with maximum loading effects on the antenna performance when the PIN diodes are integrated within the radiating patch [31]. We prefer to mount the PIN diodes and associate the biasing circuit in the ground plane instead of the patch antenna due to the biasing complexity and the antenna radiation interfering with the antenna structure [31]. The main reason for choosing three PIN diodes as switches is to change the proposed antenna effective electrical length to achieve the proposed frequency reconfigurability. The defected partial ground plane resonators self-resonance frequencies depend on their physical dimensions. The defected partial ground plane operation is very similar to a circuit with parallel inductance capacitance [8]. Through an increase in the overall length of the ground plane slot, the inductance could be raised, while reducing the slot width increases the capacitance. The notched band frequency can be approximated using equation 1 [32].

$$f_{notch} = \frac{c}{4(L\sqrt{\epsilon}_e)} \tag{1}$$

Here, the overall length of the defected partial ground plane slot is L, ϵ_e is the effective dielectric constant, and c is the light speed in the free space.
In the proposed configuration, the slot width (W) and slot length (l) are fixed to 5 mm and 1 mm, respectively. The resonant frequency can be controlled using the overall defected partial ground plane slot length through the PIN diodes. The proposed antenna depicts a dual and triple notch band only when the PIN diodes are inserted within the ground plane slot. The three PIN diodes are used to adjust inductance and capacitance of the antenna equivalent circuit. This variation contributes to the tuning of impedance matching in desired operating bands and optimizing the corresponding resonant circuitry. Figure 11(b, c, e, and d) indicates the equivalent circuits for the ON and OFF conditions.

To explain the PIN diode operation (ON-OFF), an ideal PIN diode is used as a switch in the simulation, and the proposed diodes are modelled as a metal strip only for direct open and short states. Under ON status, the respective circuit shows that the inductor (L) and a resistor (R_S) are loaded in series connections, while an inductor (L) is loaded in series with a capacitor (C_T) when the PIN is switched to OFF. In this analysis we used a suitable bias to prevent the coupling of the RF signal and the bias current [31]. Capacitors are utilized to block DC and pass RF signal. Inductors are used as chokes to block RF signal and pass DC as seen in Fig.11(d). We observed that it is possible to switch the proposed reconfigurable antenna between a UWB mode, dual band-notch modes and a triple band-notch mode. Figure 12 presents the $|S_{11}|$ spectra and the realized gain versus frequency of the proposed antenna based on the eight diode states. State 1 is establishing the UWB (1.86-10.89) GHz when all the PIN diodes are OFF. States 4, 6, and 8 establish the dual bands, and states 2, 3, 5, and 7 are solely responsible for obtaining the triple bands. The transition between dual, triple and UWB can be explained by the change in the surface current. Indeed, the electrical length of the proposed antenna structure is really determined by the diodes condition (ON and OFF). From the proposed antenna $|S_{11}|$, the antenna bandwidths are significantly affected by the PIN diode switching. It can be inferred that the parasitic element affects the antenna bandwidth to be changed from the UWB to the narrow bands. Due to parasitic effects, the proposed UWB mode is decreased to narrowband mode. The bandwidth is changed from UWB to narrow band due to the difference in surface current distribution [16]. In fact, the proposed antenna electrical length can be determined from switching PIN diodes conditions as summarized in Table II. We observed from the realized antenna gain versus frequency plot in which frequencies the UWB notch antenna radiates and where it does not. The obtained gain is found to be suitable for short and medium wireless applications [7]; ranging from 5.2 dB to 6.1 dB as shown in Fig. 12(b).

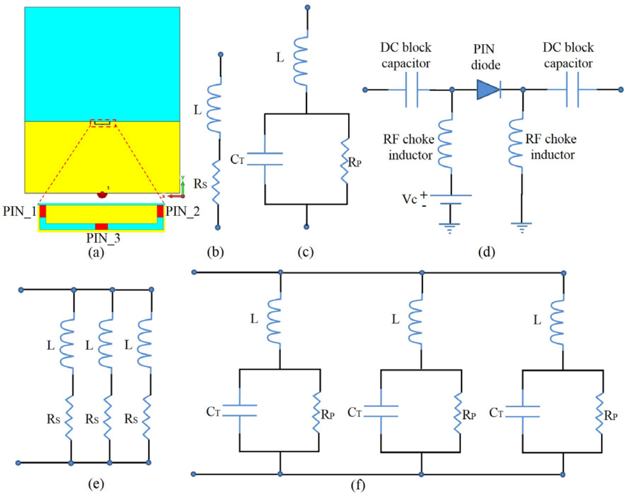


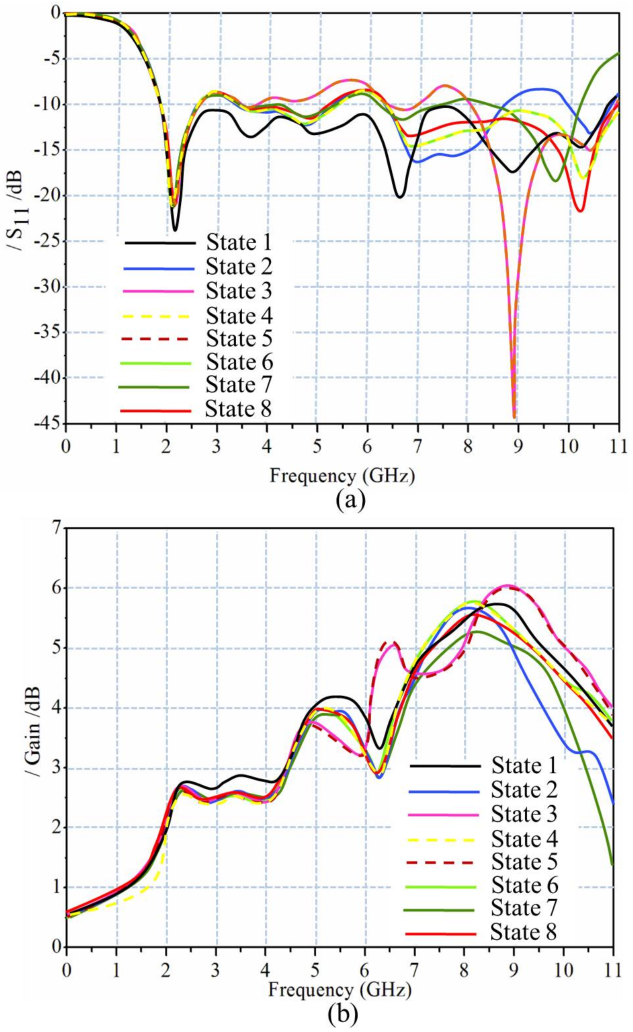
Fig. 11. The proposed PIN diodes and bias circuit (a) proposed reconfugrable antenna, (b) equivalent circuit for ON state of one PIN diode, (c) equivalent circuit for OFF state of one PIN diode, (d) bias circuit for PIN diodes, (e) equivalent circuit for ON state of all PIN diodes, and (e) equivalent circuit for OFF state of all PIN diodes,.

TABLE II
Summary of the PIN diodes switching states modes

State	PIN_1	PIN_2	PIN_3	Operation (GHz)	Peak Realized Gain (dB)
1	OFF	OFF	OFF	UWB (1.86-10.89)	5.7
2	OFF	OFF	ON	three notch (2.7-3.47), (5.45-6.4) and (8.87-10.05)	5.65
3	OFF	ON	OFF	three notch (2.76-3.53), (3.9-6.4), and (7.16-8.05)	6
4	OFF	ON	ON	two notch (2.7-3.5) and (5.43-6.44)	5.79
5	ON	OFF	OFF	three notch (2.76-3.53), (3.9-6.4), and (7.16-8.05)	6.03
6	ON	OFF	ON	two notch (2.7-3.5) and (5.43-6.44)	5.78
7	ON	ON	OFF	three notch (2.74-3.4), (5.44-6.38) and (7.54-8.4)	5.2
8	ON	ON	ON	three notch (2.7-3.5) and (5.4-6.46)	5.5

IV. Fabrication, Measurements, and Testing

The proposed antenna is fabricated calculating the optimal design. The fabrication is performed in the printed wiring board laboratory of the BME-ETT, Department of Electronic



(a)

(b)

Fig. 12. Simulated (a) reflection coefficient Vs frequency, and (b) realized gain Vs frequency for the all the PIN diodes states.

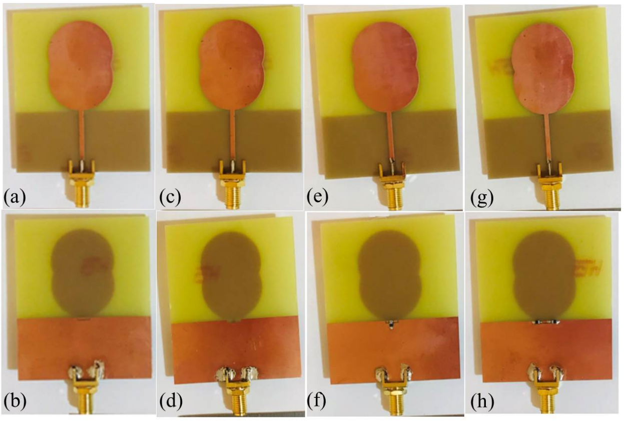


Fig. 13. The proposed antenna fabricated (a), (c) front view without PIN diodes and (e), (g) front view with PIN diodes, and (b), (d) back view without PIN diodes and (f), (h) back view with PIN diodes.

Technology, Budapest University of Technology and Economics. The Printed Circuit Board (PCB) technique is used to fabricate the proposed antenna on low-cost FR-4 substrate with a dielectric constant of 4.3, a loss tangent of 0.025, and substrate thickness of $1mm$. The fabricated antenna is fed through an SMA connector. BAR6303W PIN diodes are utilized as high-speed RF-signal switches [33]. Different shapes of the slotted ground plane connect to the PIN diodes. Four reconfigurable antennas are fabricated based on the information presented in Fig. 1. To identify the characteristics of the band notch, the PIN diodes are soldered within the slotted ground plane as shown in Fig. 13. The experimental measurements are tested using Vector Network Analyzer (VNA) STAR ms4642A Series inside an RF anechoic chamber. By using VNA ranging ($1MHz$-14 GHz), the return losses of the fabricated antenna are measured and tested to validate the simulated results with the measured ones. Due to the symmetry of the PIN diode states (2, 3, 5, and 7), we fabricated only state 3 by integrating PIN_2 within the slotted ground plane and measured the corresponding $|S_{11}|$. Because of the symmetry among states 4, 6, and 8, we fabricated only state 6 by integrating PIN_1 and PIN_3 within the slotted ground plane and measured the corresponding $|S_{11}|$ as seen in Fig. 13.

The proposed antenna is connected to a bias tee circuit through the SMA port of 50 Ω input impedance. The PIN diodes are connected to the DC source through the positive negative electrodes that come out from the bias tee to ensure no interference between the DC and the RF sources. In Fig. 14, the RF source is connected to the antenna and the bias tee electrodes are controlled by a microprocessor Arduino to control the switching process remotely during the measurements.

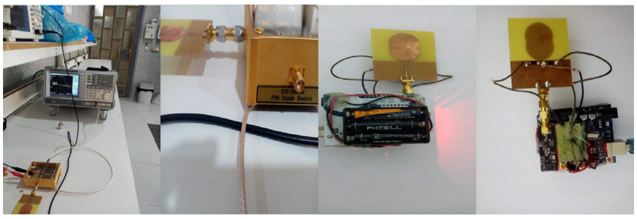


Fig. 14. Proposed antenna measurement setup, biasing circuit for PIN diode.

Antenna performance in terms of $|S_{11}|$ spectra to all the proposed swiching cases are discussed. Figure 15 shows the simulated and measured $|S_{11}|$ spectra for the proposed antenna without slots. Figure 16 shows the $|S_{11}|$ spectra for the proposed antenna with slot on the ground plane but without PIN diodes. We found that the simulated results agree very well with experimental results. The measured results almost agreed with simulated results to support that the proposed antenna provides an UWB response. There is an insignificant difference between simulated and measured results. This deviation is due to certain parameters, such as the manufacturing tolerance, the dielectric permittivity of the substrate, the simulation frequency width, the soldering conditions of the SMA connector, and the measurement circumstances [10].

Next, for the eight switching states, we discuss the first case when all three PIN diodes (PIN_1, PIN_2 and PIN_3)

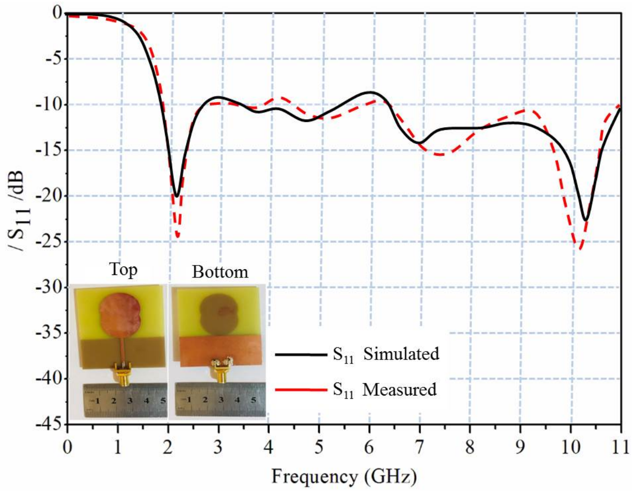


Fig. 15. Simulated and measured $|S_{11}|$ spectra of the proposed antenna without slot on the ground plane.

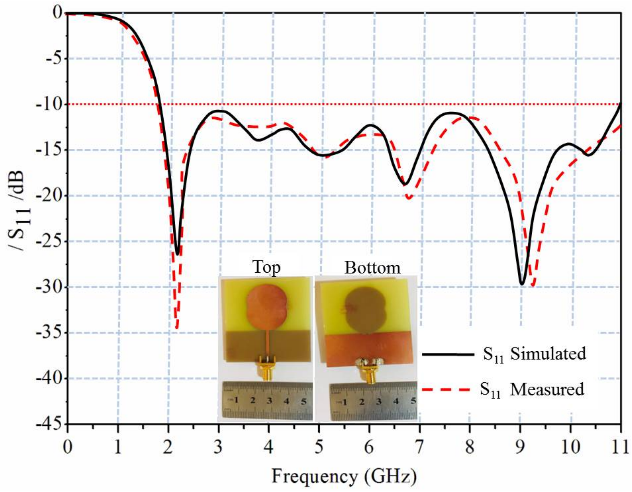


Fig. 16. Simulated and measured $|S_{11}|$ spectra of the proposed antenna with slot on the ground plane.

are in the OFF state; the proposed reconfigurable antenna operates in UWB mode with a return loss below -10 dB from (1.85-10.89 GHz). For simplicity of measurement and to prevent complexity of the connecting wire, the other PIN diodes are not soldered within the proposed antenna. We utilized a separate basis circuit (RF chock and DC block) for each PIN diode, which leads to block DC while passing RF, and RF chokes block RF while passing DC. The antenna is also linked to a Balun structure to avoid interference leakage from the coaxial cable and SMA connector. A good agreement has been obtained between the measured and simulated $|S_{11}|$ results as seen in Fig. 17.

The next state, when PIN_1 and PIN_2 are in the ON state and PIN_3 is in the OFF state, the proposed reconfigurable antenna operates in triple band mode (2.74-3.4), (5.44-6.38), and (7.54-8.4) GHz with a return loss of -23 dB, -14 dB, and -29.2 dB, respectively. If PIN_3 is turned ON and the other two PIN diodes are turned OFF the proposed reconfigurable antenna operates within three bands (2.7-3.47), (5.45-6.4), and (8.87-10.05) GHz with a return loss of -23.1 dB, -13 dB, and -18.2 dB, respectively. Figure 18 depicts the measured and simulated $|S_{11}|$ of the proposed reconfigurable antenna within the three band mode.

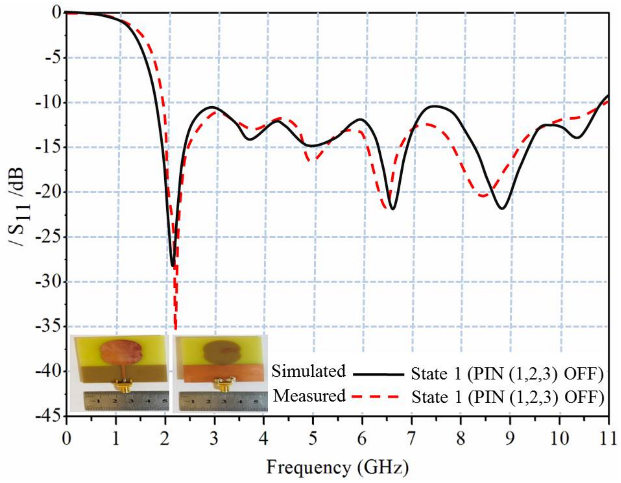


Fig. 17. The simulated and measured reflection coefficient for the operation state 1.

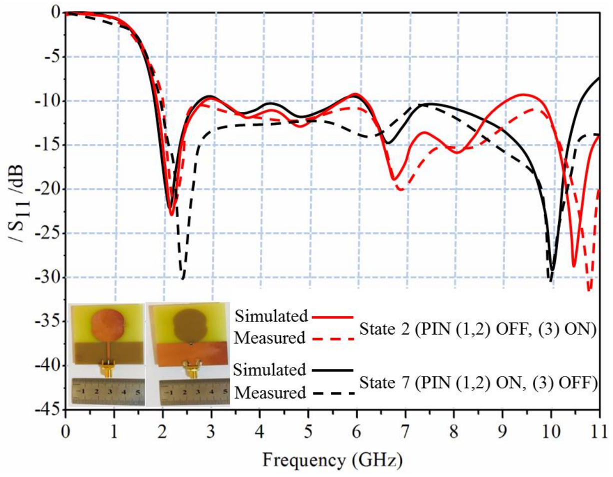


Fig. 18. The simulated and measured reflection coefficient for the operation states 2 and 7.

The next switching states when PIN_1 and PIN_3) are OFF and PIN_2 is ON, and when PIN_2 and PIN_3 are OFF and PIN_1 is ON, the proposed reconfigurable antenna operates in four band mode (1.86-2.75), (3-5.75), (6.15-6.9) and (8.86-9.75) GHz with a return loss of -23 dB, -12 dB, -13 dB and -14 dB, respectively. Figure 19 depicts the measured and simulated $|S_{11}|$ of the proposed reconfigurable antenna within the four band mode.

Another three states which have the same notch bands, state 4 when PIN_2 and PIN_3 are ON and PIN_1 OFF, state 6 when PIN_1 and PIN_3 are ON and PIN_2 OFF, and state 8 when all PIN diodes are ON, the proposed reconfigurable

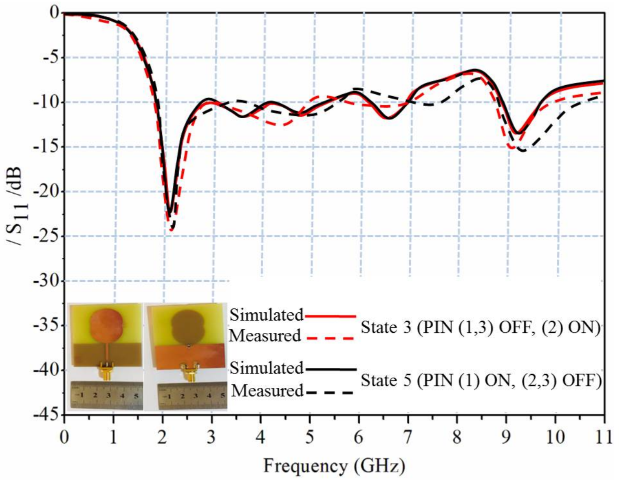


Fig. 19. The simulated and measured reflection coefficient for the operation states 3 and 5.

antenna operates within three bands (1.86-2.65), (3-5.5), and (6.15-11.9) GHz with a return loss of -23 dB, -12 dB, and -42 dB respectively. Figure 20 depicts the measured and simulated $|S_{11}|$ of the proposed reconfigurable antenna within the three band mode.

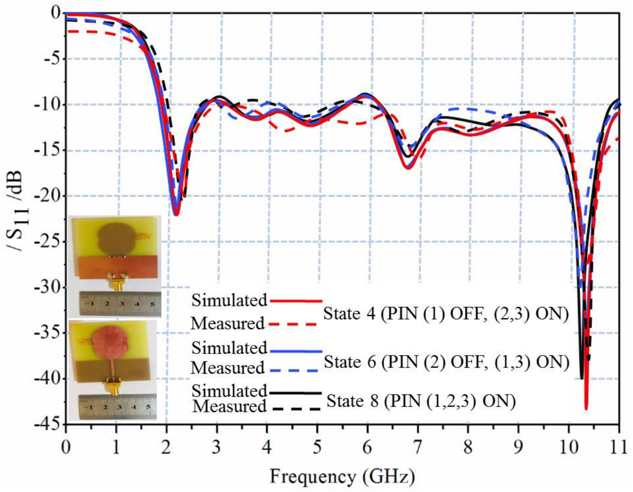


Fig. 20. The simulated and measured reflection coefficient for the operation states 4, 6 and 8.

From all the previous simulated and measured results, it is indicated that the proposed antenna has good ability to switch among UWB mode, dual, triple, and four band modes. The antenna radiation patterns are measured inside an RF chamber using a three-antenna method. The process is basically realized by conducting a standard dipole antenna at the frequency band of interest after calibrating the path losses inside the chamber. The simulated and measured radiation patterns in the E-plane and H-plane are depicted in Fig. 21 at 2.5 GHz. It is clear the radiation patterns of the proposed antenna display the same characteristics which are almost symmetrical in the E-plane and omnidirectional in the H-plane for the eight PIN diode states. Such stability in the radiation patterns for all the switches states in both E-plane and H-plane is a very desirable property in many applications [34]. This antenna patterns stability is due to selecting the PIN diodes position on the ground plane instead of the radiating patch element. However, due to manufacturing tolerance and the additional parasitic parameters caused by introducing several capacitors and bias wires, the simulated and measured radiation patterns are worse when integrating the PIN diodes on the antenna batch instead of the ground plane [34]. These eight radiation patterns under different biasing states exhibited certain similarities to each other, which confirmed the purpose of the proposed frequency reconfigurable antenna. Further, we noticed that for the eight switching states, the radiation pattern in the H-plane has an ”8” shape, which indicates bidirectional radiation. This pattern is a typical radiation pattern like a conventional monopole antenna. The proposed reconfigurable antenna can have many possible applications in modern UWB and multi-functional mobile communication systems due to its excellent performance, ease of control and modification, and simple structure.

The current reconfigurable antenna is compared with the previously reported work [17]-[25]. Table III summarizes the comparison between this work and the literature. The proposed reconfigurable antenna has a very wide bandwidth as well as smaller size than [23]. The current design used only three PIN diodes compared with [23] and [25] for similar performances, which employs more than three PIN diodes. Another feature of the proposed antenna can switch among the UWB, dual, and triple band mode compared to the proposed antenna in [17]-[25], designed within a single mode. Thus, the present reconfigurable UWB antenna is more easily controlled among the desired band. The notch band characteristics can be realized by changing the PIN diode status rendering the antenna appropriate for UWB, dual, and triple operating bands. The low design complexity is another achievement in this work compared with suggested antenna design in [17]-[25]. The proposed antenna design does not have any slot on the radiation patch, resulting in UWB having notched band characteristics, therefore, the proposed technique does not significantly affect the proposed antenna’s features. Finally, the proposed antenna performances unaffected due to the low ON resistance and insertion loss, as well as the PIN diode’s easy biasing approach.

V. Conclusion

In this paper, we presented a compact UWB antenna that is low cost, light weight, and easy to control with an overall size of $50 \times 60 \times 1$ mm^3. The proposed antenna employs mounted three PIN diodes on the ground plane for achieving notched band characteristics. The reconfigurable antenna is fabricated on low-cost FR-4 substrate with ϵ_r = 4.4. It is designed to cover the entire UWB spectrum from 1.7 GHz to 11 GHz. The reconfigurability of the frequency is accomplished by adjusting the effective electrical length of the proposed slot antenna. The reconfigurability is based on the insertion of three PIN diodes placed within a slotted ground plane. Due to the simplicity of the biasing circuit, it is simple to select

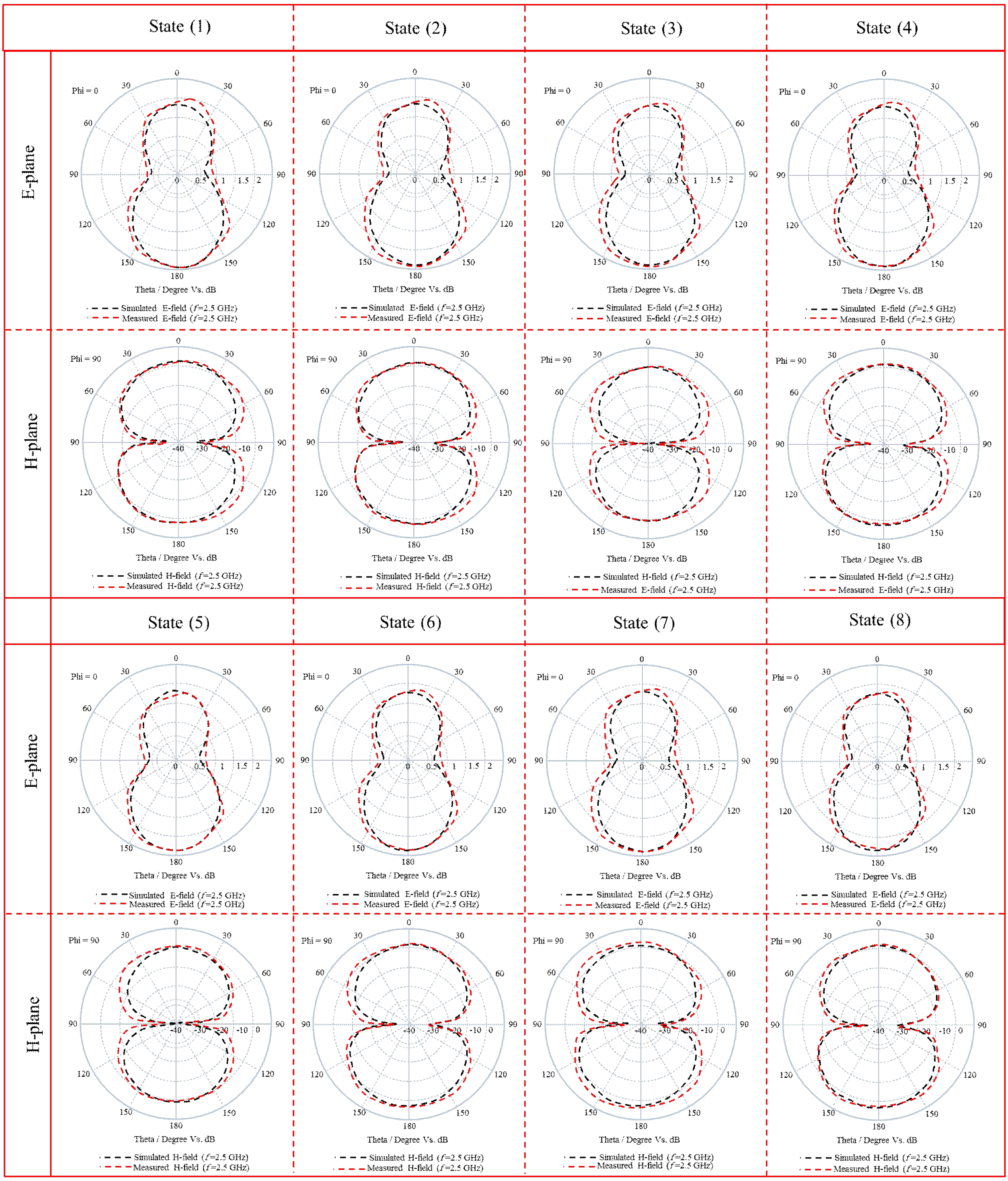


Fig. 21. Simulated and measured far-field radiation patterns at 2.5 GHz in the E-plane and H-plane of the proposed UWB band notched antenna with all PIN diodes states.

TABLE III
COMPARISON BETWEEN THE PROPOSED WORKS WITH RESPECT TO OTHER PUBLISHED RESULTS.

Reference	**Size** (mm^2)	**Reconfiguration**	**Substrate**	**RF switch**	**Notch bands**	**Rejected bands**(GHz)	**No. of PIN diodes**	**Design Complexity**
[17]	30× 30	N	FR4	Varactor	N	(2.7-7.1)	1	Moderate
[18]	48× 7.47	Frequency	TLY-3-0450C5	MEMS	N	(4.8-7.4)	N	Moderate
[19]	10× 12	Frequency	RO4350B	Varactor	Dual	1.2-1.65	1	High
[20]	50× 50	Frequency	FR4	FET	N	2.4,3.3,4.2,5.4	N	High
[21]	33× 16	Frequency	FR4	N	single/dual	2.1,2.4,3.5,4.15,4.8,5.2	N	Moderate
[22]	50× 50	Pattern	RO3006	PIN	N	5.2,5.8,6.4	2	High
[23]	100× 100	Polarization	RT/duroid 5880	PIN	Dual	2.53,2.44	5	High
[25]	25× 25	Frequency	FR4	PIN	Miltiband	3.85,4.14,4.43,4.91,6.01	5	High
This work	50× 60	Frequency	FR4	PIN	Single, dual, triple, UWB	(2.77-3.3), (5.51-6.25) and (10.87-20)	3	Low

the position to connect the biasing circuit. There is little impact on antenna bandwidth, gain, and radiation efficiency. The proposed PIN diode as a switching device is mounted within the ground plane instead of the antenna patch body. The main reason to choose three PIN diodes as switches is to change the proposed antenna's effective electrical length to achieve frequency reconfigurability. By switching the state of the PIN diode changes between forward and backward, you may get a double, triple, or UWB band mode. When all the switches are OFF, the proposed antenna is in a UWB operation band (1.85-10.9) GHz. The proposed antenna operates in dual band mode 2.7-3.5 GHz and 5.4-6.46 GHz, when all switches are ON. The other six diode conditions allow for the operating band 2.7-3.5, 5.4-6.4, 7.5-8.4, and 8.87-10.05 GHz. The conventional monopole antenna is operating over the UWB. In this paper, the proposed antenna based on the PIN diodes is simulated, fabricated, measured, and tested. The simulated and measured results agree with the terms of $|S_{11}|$ spectra and radiation patterns for the proposed UWB antenna. The inclusion of the PIN diodes for the frequency notching on the antenna ground plane without impacting the antenna performance, which allowed more flexibility in the design of our antenna. The proposed reconfigurable monopole antenna with UWB, dual and triple operating bands is a good candidate for wireless applications such as WiFi and WiMAX. The proposed antenna could serve as an effective multimode application such as UWB and military reconfigurable frequency antennas.

ACKNOWLEDGMENT

The research reported in this paper was supported by the BME NC TKP2020 grant of NKFIH Hungary. The author Yahiea Al Naiemy would like to record his indebtedness to Tempus Public Foundation Stipendium Hungaricum program and University of Diyala – Iraq for their support for his PhD scholarship.

REFERENCES

[1] Row, Jeen-Sheen;You-Heng, Wei. Wideband reconfigurable crossed-dipole antenna with quad-polarization diversity. *IEEE Transactions on Antennas and Propagation* 2018, 66.4, 2090–2094. DOI: 10.1109/TAP.2018.2800785

[2] Ojaroudi, Parchin; Naser, et al. Recent developments of reconfigurable antennas for current and future wireless communication systems. *Electronics* 2019, 8.2, 128. DOI: 10.3390/electronics8020128

[3] Tang, Ming-Chun; et al. Compact, frequency-reconfigurable filtenna with sharply defined wideband and continuously tunable narrowband states. *IEEE Transactions on Antennas and Propagation* 2017, 65.10, 5026- 5034. DOI: 10.1109/TAP.2017.2736535

[4] Alnaiemy, Yahiea; Lajos, Nagy. Design of a Controllable Antenna Based on Embedded Differential PSK Modulation. *Progress In Electromagnetics Research* 2021, 90, 43-62. DOI: 10.2528/PIERB20100106

[5] Majid, Huda A.; et al. Frequency and pattern reconfigurable slot antenna. *IEEE transactions on antennas and propagation* 2014, 62.10, 5339-5343. DOI: 10.1109/TAP.2014.2342237

[6] Zhu, Ziqiang; et al. A flexible frequency and pattern reconfigurable antenna for wireless systems. *Progress In Electromagnetics Research* 2018, 76.10 , 63-70. DOI: 10.2528/PIERL18040401

[7] Han, Liping; et al. Design of frequency-and pattern-reconfigurable wideband slot antenna. *International Journal of Antennas and Propagatio*n 2018, 1-7, 2018. DOI: 10.1155/2018/3678018

[8] Iqbal, Amjad, et al. Frequency and pattern reconfigurable antenna for emerging wireless communication systems. *Electronics* 2019, 8.4, 407. DOI: 10.3390/electronics8040407

[9] Zhang, Zhuohang; Zhongming, Pan. Time domain performance of reconfigurable filter antenna for IR-UWB, WLAN, and WiMAX applications. *Electronics* 2019, 8.9, 1007. DOI: 10.3390/electronics8091007

[10] Wu, Terence; et al. Switchable quad-band antennas for cognitive radio base station applications. *IEEE Transactions on Antennas and Propagation* 2010, 58.5, 1468-1476. DOI: 10.1109/TAP.2010.2044472

[11] Hamid, M, R.; et al. Switched-band Vivaldi antenna. *IEEE transactions on antennas and propagation*. 2011, 59.5, 1472-1480. DOI: 10.1109/TAP.2011.2122293

[12] Yadav, Ajay; Minakshi, Tewari; Rajendra, Prasad, Yadav. Pixel shape ground inspired frequency reconfigurable antenna. *Progress In Electromagnetics Research*. 2019, 89, 75-85. DOI: 10.2528/PIERC18082102

[13] Tasouji, Nasrin; et al. A novel printed UWB slot antenna with reconfigurable band-notch characteristics. *IEEE Antennas and wireless propagation letters*. 2013, 12, 922-925. DOI: 10.1109/LAWP.2013.2273452

[14] Han, Liping; Jing, Chen; Wenmei, Zhang. Compact UWB monopole antenna with reconfigurable band-notch characteristics. *International Journal of Microwave and Wireless Technologies*. 2020, 12.3, 252-258. DOI: 10.1017/S1759078719001296

[15] Oraizi, Homayoon; Nooshin, Valizade Shahmirzadi. Frequency-and time-domain analysis of a novel UWB reconfigurable microstrip slot antenna with switchable notched bands. *IET Microwaves, Antennas and Propagation*. 2017, 11.8, 1127-1132. DOI: 10.1049/iet-map.2016.0009

[16] Li, Yingsong, et al. A small multi-function circular slot antenna for reconfigurable UWB communication applications. *IEEE Antennas and Propagation Society International Symposium (APSURSI)*. 2014, 1, 834- 835. DOI: 10.1109/APS.2014.6904745

[17] Aghdam; Sajjad, Abazari. A novel UWB monopole antenna with tunable notched behavior using varactor diode. *IEEE Anten- nas and Wireless Propagation Letters*. 2014, 13, 1243-1246. DOI: 10.1109/LAWP.2014.2332449

[18] Saghati; Alireza, Pourghorban; Kamran, Entesari. A reconfigurable SIW cavity-backed slot antenna with one octave tuning range. *IEEE transactions on antennas and propagation*. 2013, 61.8, 3937-3945. DOI: 10.1109/TAP.2013.2263215

[19] Behdad, Nader; Kamal, Sarabandi. A varactor-tuned dual-band slot antenna. *IEEE Transactions on Antennas and Propagation*. 2006, 54.2, 401-408. DOI: 10.1109/TAP.2005.863373

[20] Aboufoul, Tamer; Akram, Alomainy; Clive, Parini. Reconfiguring UWB monopole antenna for cognitive radio applications using GaAs FET switches. *IEEE Antennas and Wireless Propagation Letters*. 2012, 11, 392-394. DOI: 10.1109/LAWP.2012.2193551

[21] Shah, I. A.; et al. Design and analysis of a hexa-band frequency reconfigurable antenna for wireless communication. *AEU-International Journal of Electronics and Communications*. 2019, 98, 80-88. DOI: 10.1016/j.aeue.2018.10.012

[22] Nikolaou, Symeon; et al. Pattern and frequency reconfigurable annular slot antenna using PIN diodes. *AEU-International Journal of Electronics and Communications*. 2006, 54.2, 439-448. DOI: 10.1109/TAP.2005.863398

[23] Kim, Boyon; et al. A novel single-feed circular microstrip antenna with reconfigurable polarization capability. *IEEE Transactions on Antennas and Propagation*. 2008, 56.3, 630-638. DOI: 10.1109/TAP.2008.916894

[24] Elwi, Taha A. Remotely controlled reconfigurable antenna for modern 5G networks applications. *Microwave and Optical Technology Letters*. 2020. DOI: 10.1002/mop.32505

[25] Singh, Prem Pal; et al. Frequency reconfigurable multiband antenna for IoT applications in WLAN, Wi-Max, and C-band. *Progress In Electromagnetics Research*. 2020, 102, 149-162. DOI: 10.2528/PIERC20022503

[26] Reddy, Bobbili Naga Balarami; et al. Design and analysis of wideband monopole antennas for flexible/wearable wireless device applications. *Progress In Electromagnetics Research M*. 2017, 62, 167-174. DOI: 10.2528/PIERM17092107

[27] Mohandoss, Susila; et al. Fractal based ultra wideband antenna development for wireless personal area communication applications. *AEU International Journal of Electronics and Communications*. 2018, 93, 95- 102. DOI: 10.1016/j.aeue.2018.06.009

[28] Kingsly, Saffrine; et al. Multiband reconfigurable filtering monopole antenna for cognitive radio applications. *IEEE Antennas and Wireless Propagation Letters*. 2018, 17.8, 1416-1420. DOI: 10.1109/LAWP.2018.2848702

[29] Mohandoss, Susila; et al. On the bending and time domain analysis of compact wideband flexible monopole antennas. *AEU-International Journal of Electronics and Communications*. 2019, 101, 168-181. DOI: 10.1016/j.aeue.2019.01.015

[30] Studio, CST Microwave. URL http://www.cst.com/Content/Products/MWS.. *Overview*. aspx. 2017.

[31] Rahim, M. K. A.; et al. Frequency reconfigurable antenna for future wireless communication system. *2016 46th European Microwave Conference (EuMC)*. IEEE 2016, 2016. DOI: 10.1109/EuMC.2016.7824506

[32] Lakrit, Soufian; et al. Compact UWB flexible elliptical CPW-fedantenna with triple notch bands for wireless communications. *International Journal of RF and Microwave Computer-Aided Engineering*. 2020, 30.7, 2020. DOI: 10.1002/mmce.22201

[33] www.alldatasheet.com, datasheetpdf. URL http://www.alldatasheet.com/INFINEON./396627,BAR63-02V.

[34] Al Gburi, Mohaimen; and Muhammad Ilyas. A Novel Design Reconfigurable Antenna Based on the Metamaterial for Wearable Applications. *Journal of Physics: Conference Series*. 2021, Vol. 1973. No. 1. IOP, 2021. DOI: 10.1088/1742-6596/1973/1/012042

Yahiea Alnaiemy He received his Bachelor's Degree in Electrical Engineering from AL-Mustansiriyah University, Faculty Of Engineering in 1998. He continued his graduate studies by joining the Iraqi Commission for Computers and Informatics, where he received a Higher Diploma in Information Systems in 2001. He enrolled at Diyala University as an instructor in communication Engineering, electrical power, computer, and physics departments. In 2009, he granted a scholarship to complete his master's degree in electrical engineering at the University of Arkansas at Little Rock, USA. He got his MSc in Wireless Communications from UALR, USA, in 2012. While completing his graduate degree, his research effort has been in the area of antennas and microwave material characterization. In 2017, he granted a scholarship to complete his Ph.D. in electrical engineering at BME, Hungary. His current research areas include UWB antennas, EBG structures, metamaterial, GPS, implantable wireless systems, nanoscale microwave devices, reconfigurable antennas, and RF power harvesting. He has been an IEEE Member and reviewer in several international journals and conferences since 2011.

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

Demonstrating BB84 Quantum Key Distribution in the Physical Layer of an Optical Fiber Based System

Márton Czermann[1,4], Péter Trócsányi[1], Zsolt Kis[2], Benedek Kovács[3] and László Bacsárdi[1] *Member, IEEE*

***Abstract*—Nowadays, widely spread encryption methods (e.g., RSA) and protocols enabling digital signatures (e.g., DSA, ECDSA) are an integral part of our life. Although recently developed quantum computers have low processing capacity, huge dimensions and lack of interoperability, we must underline their practical significance – applying Peter Shor's quantum algorithm (which makes it possible to factorize integers in polynomial time) public key cryptography is set to become breakable. As an answer, symmetric key cryptography proves to be secure against quantum based attacks and with it quantum key distribution (QKD) is going through vast development and growing to be a hot topic in data security. This is due to such methods securely generating symmetric keys by protocols relying on laws of quantum physics.**

In this paper we introduce a fiber based QKD system that is being built in Hungary in a collaboration between Budapest University of Technology and Economics (BME), Wigner Research Centre for Physics and Ericsson Hungary. We demonstrate the first successful quantum key distribution over physical layer in accordance with the truth table of BB84 protocol in the country. We apply light pulses at 1550 nm wavelength, reducing their power to less than one photon per pulse level. We create two phases of operation including an initialization phase in which software and hardware solutions are proposed for synchronizing the units of the two communicating parties. We introduce a data processing and a timing mechanism and elaborate on the results of the demonstration. We also inspect the possibilities of efficiency enhancement and give an outlook on further development directions.

***Index Terms*—BB84; quantum key distribution; symmetric key encryption; phase encoding; fiber optic system; adaptive filtering; synchronization**

I. Introduction

We need encryption to send data securely. Today's encryption solutions can be divided into two major groups: symmetric and asymmetric key encryption. However, quantum computers pose serious threats on asymmetric key encryption due to Shor's algorithm, which is a polynomial-time quantum computer algorithm for integer factorization. But there are symmetry-key algorithms like One-Time-Pad (OTP) which provides mathematically proven security. The critical question is how the communicating parties can share the key used for symmetric encryption since they need to use the same key for both encryption and decryption [1], [2].

Quantum key distribution (QKD) [3], [4] offers an efficient and secure solution for this key exchange and its security is based on the laws of physics. Since unknown quantum bits cannot be copied due to the No Cloning Theorem (NCT) [5], an attacker does not have the opportunity to copy information which is being shared between Alice and Bob. This means that a passive attack is not possible against QKD protocols, the eavesdropper must actively intervene. However, QKD protocols work in such a way that the active presence of an eavesdropper disrupts communication, bringing noise into the quantum channel which can be detected by the communication parties. So the presence of an attacker would be revealed.

At Budapest University of Technology and Economics, we've already researched QKD both in theory [6] and practice [7] in the recent years as well as researched different quantum random number generator setups [8]. In this paper, we present the demonstration results of the first Hungarian QKD system which uses BB84 protocol for key exchange. The article is organized as follows. Section II gives a short overview of different QKD initiatives. Section III introduces our system, while Section IV details our procedures used for initializing signal levels and timing. The operation of the system is described in Section V, our demonstration results in Section VI.

II. QKD in the 21st century

Although the majority of quantum links and networks established during the past two decades have been fiber-based, there have been several examples of free-space approaches, too. Since the technology of optical telecommunications is widespread, applying attributes of light as quantum carrier is a favorable solution. Having the infrastructure already deployed facilitated the development of terrestrial fiber-based quantum links and networks implementing various QKD protocols – such as the 3.6 Tbps optical backbone network deployed by China United Telecom in which a quantum communication (QC) link was integrated with classical ones [9].

The prospect of scalability, however, is challenging for this kind of key distribution due to the attenuation of fibers, which limits the transmission range of photons to a few hundred

[1] Department of Networked Systems and Services, Budapest University of Technology and Economics, Magyar tudósok krt 2., Budapest 1117, Hungary. (e-mail: czermann@mcl.hu, p.trocsanyi@edu.bme.hu, bacsardi@hit.bme.hu)

[2] Wigner Research Centre for Physics, Budapest, Hungary (e-mail: kis.zsolt@wigner.hu)

[3] Ericsson Hungary, Budapest, Hungary (e-mail: benedek.kovacs@ericsson.com)

[4] corresponding author

This research was supported by the National Research Development and Innovation Office of Hungary within the Quantum Technology National Excellence Program (Project No. 2017-1.2.1-NKP-2017-00001) and by the Ministry of Innovation and Technology and the National Research, Development and Innovation Office within the Quantum Information National Laboratory of Hungary.

DOI: 10.36244/ICJ.2021.3.5

kilometers even applying the best quality optical fibers. Other terrestrial solutions are based on trusted nodes that must either be physically secured in order to be considered as a segment of a secure key distribution method or apply quantum repeaters. The latter option requires the complex technology of quantum memories in order to find a workaround for the NCT and extend the distances of QKD. In spite of scepticism around the practicability of this late technology, various architectures based on space-borne quantum memories have already been proposed [10]. Recent study on quantum technologies in space in general has also been carried out, summarizing the state of the art of this area [11]. Satellite-based quantum communication and technologies [12], [13], [14], [15] provide the means of bridging terrestrial fiber-based metropolitan networks and free-space links thus offering a scalable solution for physically secure communication and indicate directions towards a future global quantum internet [16], [17], [18].

Classifying the QKD systems by the protocols implemented by them, there are two fundamental protocol families that we can exemplify: prepare and measure and entanglement based. Both of them provide security for key distribution grounded in the laws of quantum mechanics. While former type of protocols operate with initial keys and exploit NCT, the latter ones are built on a quantum phenomenon: entanglement [5]. Although the vast majority of prepare and measure protocols is based on the widely-known BB84 protocol [3], several links have been established exploiting entanglement in the past two decades [19], [20], [21], [22], [23], [24]. Since we performed the demonstration on a prepare and measure system we would like to introduce some milestones from this family to make the navigation on the map of QC easier.

The first operating fiber quantum network was built in 2003 in the USA as a project of the Defense Advanced Research Projects Agency (DARPA) [25]. QKD was put into practice via coherent laser pulses propagating between 4 nodes to which a further free-space link with 2 extra nodes was connected later. In 2004 Austria launched a 4-year project called Secure Communication based on Quantum Cryptography (SECOQC), which facilitated the establishment of a quantum network with 6 nodes investigating the operation of 8 different protocols implemented between them [26]. SECOQC also proposed the idea of multi-layer QKD networks, such as SwissQuantum [27], which implements a structure similar to the one presented in Vienna. Three layers are applied for communication: a quantum layer, a key management layer and an application layer. During a 21 month operation between 2009 and 2011 low quantum bit error rate (QBER) was stable generating 300-900 thousand symmetric secret keys on a daily basis. In 2010 Tokyo also established their own project with the aim of observing the properties of a metropolitan quantum key distribution network with trusted nodes [28].

Finally, we can not enlarge upon QC without mentioning the achievements of China. In 2016 a satellite called Micius was launched, which connected 3 ground stations (Graz in Austria, Xinlong and Nanshan in China) as a trusted relay in an intercontinental QKD setup, generating symmetric keys at a distance of 7200 km between the two countries [29]. By 2018, a 2000 km long quantum key distribution link between Beijing and Shanghai was established connecting 32 nodes involving 4 metropolitan quantum networks. Finally, a paper in January of this year introduced an integrated space-to-ground quantum communication network over 4,600 kilometres based on the previously mentioned achievements [30].

Besides experimental quantum links and networks there are several concepts and objectives for QKD applications in real-life scenarios as well [31], [32], [33], [34], [35]. The accelerating tendency in the developement of QC (and technologies) has ushered in the era of the so called 'second quantum revolution.' The expectations of this era include on behalf of the European Union to drive their own technological improvements in a global direction. Therefore the European Commission established an initiative called EuroQCI with the promise of the deployment of a Europe-wide QC network [36].

III. The BB84 system with phase coding

Our project is based on an architecture proposed in a 2002 publication [37]. The fiber-optic QKD link implements a flavor of the BB84 protocol that operates with very weak, phase modulated light pulses as quantum carrier between Alice and Bob. The light source and single photon avalanche detectors (SPADs) are both on Bob's side and part of a large-scale interferometer in which the emitted pulses propagate from Bob to Alice and back. After they return the mean photon number is less than one per pulse. The phase shifts on the pulses are assigned to classical 0 / 1 values in initial keys randomly generated by Alice and Bob and after the interference classical 0 / 1 values are assigned to detection ticks of SPADs on the two output channels of the interference as symmetric raw key bits. The design of the system and development on it had been carried out in the spirit of plug & play operability: initialization and system operation should become automated and not need user monitoring.

The physical realization can be subdivided into a quantum transmitter and a receiver side, respectively A and B, for Alice's and Bob's units, in the schematic seen in Figure 1. The photon pulses that are emitted at 1550 nm wavelength and 5 MHz repetition rate travel back and forth between Bob and Alice. Bob sends altogether 480 pulses of 20 ns width called one frame. After they are emitted, the pulses go through an ATT and an inline polarizer (ILP) before they arrive to the CIR. Since the proper pulsed operation of our laser requires the generation of high intensity pulses, we inserted an attenuator right after the source in order to protect our SPADs from damage caused by crosstalk between CIR ports. Furthermore, the generation of short pulses requires a small level of bias current in the dark period of laser diode, which in turn results in some under threshold faint glowing of the laser diode, which yields quite high level of ambient photons. This emission increases the noise background of the photon counters. The attenuator suppresses efficiently this background. The ILP enhances the degree of polarization of the laser light. For the protection of our source the CIR proves to be a suitable solution as returning pulses are directed away from it (into SPAD1).

A 50 / 50 beam splitter (BS) creates the two separate arms of the interferometer: leaving the circulator the pulses get split

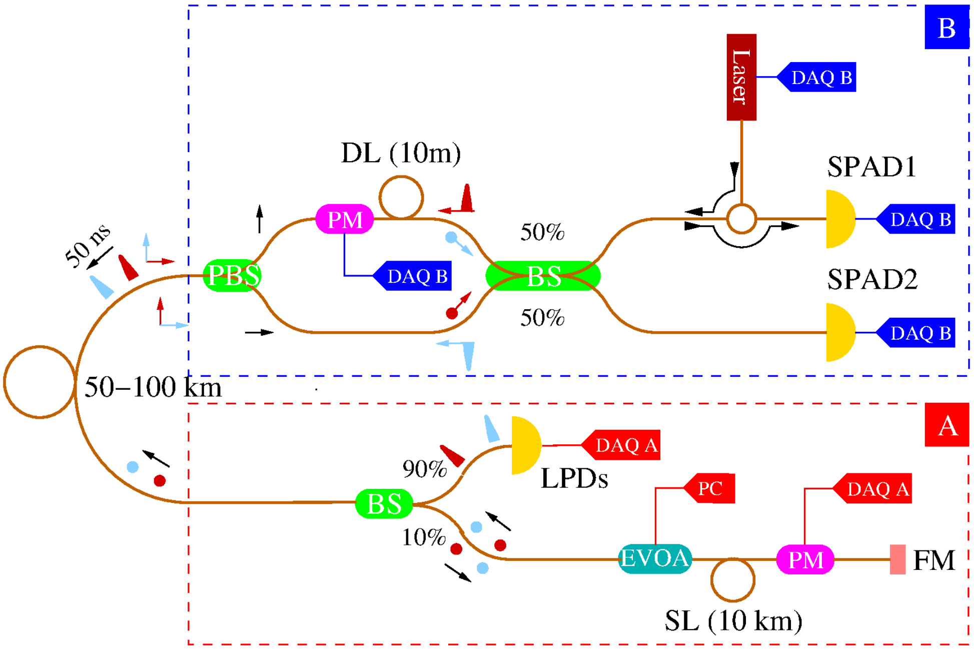


Fig. 1. Schematic of the QKD system as separate units of communicating parties.
Bob's unit (B): high power pulses are routed by a circulator (CIR) to the input port of a balanced beam splitter (BS) creating a reference (blue) and signal (red) pulse, marked with humps, and a polarization beam splitter (PBS) joins the signal paths of these. Alices's unit (A): first, the pulses are split by a 90 / 10 BS into a high power trigger signal for a linear photodetector (LPD) and low power carrier marked with dots, then the latter propagates through an attenuator (ATT), a storage line (SL) and a lithium niobate electro-optic phase modulator (PM) and back upon reflection on a Faraday Mirror (FM), and meanwhile the carrier is attenuated to the quantum level as it leaves Alice. As the carrier pulses return to Bob, their polarizations are switched, denoted by perpendicular arrows of the respective colors. The polarization switching lets swap paths propagating back and forth between Alice and Bob. According to their interference signal they produce ticks in SPAD1 or SPAD2 with given probabilities. Here data acquisition (DAQ) marks any point where a higher layer interface is needed.

into pairs here. Half of each pair gets a 50 ns time delay passing through a 10 m long delay line (DL) that is followed by Bob's phase modulator (PMb) which is off at this stage PMb. The two arms are coupled into a PBS that results in the delayed half sustaining a polarization orthogonal to the leading half. From this point on the 480 pairs follow each other to Alice's unit with this 50 ns time delay between the rising edges of the halves.

Traveling through a 50-100 km long optical fiber they arrive to Alice's side. During this experimental phase of the project we inserted only a 50 m long fiber for simplicity. Here, a 90 / 10 BS deflects the majority of the power into a linear photodetector (LPD) which has a central role in the timing mechanism for Alice's modulation as a monitor point for the position of incoming frames. The leading half of each pulse pair provides us the reference used later for the interference (when arriving back to Bob) and we modulate every trailing one. Alice performs encoding by phase shifting in either $(0-\pi)$ or $(\pi/2-3\pi/2)$ basis based on two bit random sequences as initial keys. The next component for the pulses is an electronic variable optical attenuator (EVOA), which we can control by software to reduce the pulse energy level to around one-photon after they get reflected from a Faraday Mirror (FM) and start their way back to Bob.

One last component on the transmitter side is a 10 km long storage line (SL) between the EVOA and Alice's phase modulator (PMa). This fiber segment is long enough to 'store' all of the 480 pulse pairs. If we think of a scenario without an Alice-side SL, the interaction of the forward and backward propagating pulse trains takes place between Alice and Bob in the multiple km long connection fiber. Regarding optical power, forward travelling pulses have high intensity (to be detectable with the sensitivity of the LPDs) while backward only one-photon level ones propagate. The Rayleigh scattering from the higher level pulses adds false detection to the backward travelling pulse train and higher noise level at the receiver. Inserting the SL prevents this scenario from occurring.

Returning to Bob's side the pairs are routed to opposite ports of the PBS than when they have left due to the FM swapping their polarization. This way the delayed halves take the shorter path, while the reference halves choose the path with the DL and PMb. Here, Bob modulates the phase randomly choosing from the basis $(0-\pi)$ based on his initial binary key. As all of the fibers are polarization maintaining on Bob's side and the optical path to Alice and back is identical for corresponding pulses, they arrive simultaneously to the balanced BS with identical polarization. If both parties choose the same basis, the pulse arrives to SPAD2 when the pulse pairs are in phase and to SPAD1 if one of them has a π phase shift. The interference is not deterministic in the case of different basis choices so then we will discard our classical bits assigned to the detector signal. Finally, Alice and Bob performs the basis reconciliation over Ethernet to get their symmetric sifted key.

We have always been taking plug & play principles into consideration as well as compact construction and economic but efficient operation. Still, until reaching the phase of a possible deployment the project serves scientific and research interests. Alice and Bob are connected to a power supply for the laser, mixed signal digital oscilloscope (MSO) for data acquisition (DAQ) and two 16-bit arbitrary waveform generator cards inside separate computers responsible for controlling the optical components. There is one more software-controllable voltage source for EVOAs.

IV. INITIALIZING SIGNAL LEVELS AND TIMING

We can describe two distinct phases of operation: initialization and key generation. In a deployed and working system the initialization phase sets attenuation levels and timing parameters to prepare for key generation so it's used less frequently. To enable quantum security the optical signal transmitted by Alice must be at the <1 photon / pulse level and for the optimal effect of phase modulation and noise rejection precise timing of the pulses is needed. Fluctuations in the physical properties of the fiber used as quantum channel need to be compensated for to control these conditions and the initialization tasks described here can be repeated as needed to achieve this. They include: setting attenuation levels, measuring the optical length of Alice's side, finally, based on several thousand frames setting up the series of arrival times of every incoming pulse. For this we developed our own algorithm aiming to achieve a fully automatic initialization phase, in accordance with the plug & play principles.

A. *Setting optical power levels*

We require different optical power levels for the two operation phases because we alternate between using different types of detectors. During initialization we need higher optical power level for our self-designed LPDs on Alice's side (while measuring optical path length, see in next step) than during key distribution.

The optical power received by Alice is increased by switching only Alice's attenuation. The increase is not as critical as the base power level for the key generation phase, it's just enough for both of Alice's LPDs to produce a signal high enough above the detectors' internal noise level to trigger the oscilloscope.

B. *Measuring Alice's optical path length*

We determine the instantaneous propagation time with 2 ns definition still using the oscilloscope. The bulk of Alice's signal path is the SL, in case of which our method for the length measurement exploits the role of the two LPDs. The first (LPDc) detects frames arriving to Alice (deflected by the 90 / 10 BS), while the other (LPDm) detects them travelling backwards (after reflection on the FM). Following the optical path of a single pulse entering Alice's device, we can determine the round trip time of the pulse when it travels back and forth between the BS and FM.

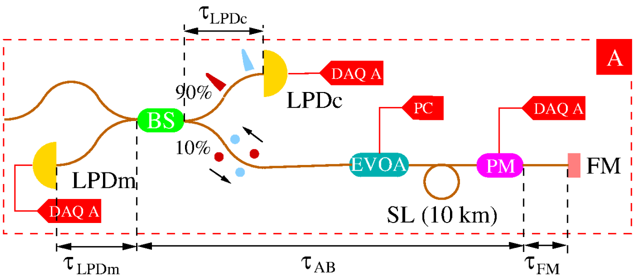


Fig. 2. More detailed view of Alice's components. Complementing the LPD on the signal input port of the BS another one was inserted on the tap output port. Notations: $\tau_{\rm LPDc}$ denotes the time delay between the BS and LPDc, while $\tau_{\rm LPDm}$ is the same for LPDm, $\tau_{\rm AB}$ is the time delay between BS and PM, while $\tau_{\rm FM}$ is the time delay between the PM and FM.

1) *Details of the calculation:* Using the notation of Figure 2, the delay time $T_{\rm total}$ between the two LPDs' signals (which detect identical parts of the incoming signal split at BS) is

$$\begin{aligned} T_{\rm total}=&2(\tau_{\rm AB}+\tau_{\rm FM})-\tau_{\rm LPDc}+\tau_{\rm LPDm}\\ =&2(\tau_{\rm AB}+\tau_{\rm FM}), \end{aligned} \tag{4.1}$$

where the two time delays $\tau_{\rm LPDc}$ and $\tau_{\rm LPDm}$ cancel, since the fiber lengths are the same between the BS and the two LPDs. We have mentioned in Section III that the trailing part of each pulse pair should be modulated. The minimal delay $T_{\rm min}$ between the detection of the leading part of an incoming pulse pair by LPDc and the modulation of the trailing part of the same pulse pair is given by

$$T_{\rm min}=\tau_{\rm AB}-\tau_{\rm LPDc}+2\tau_{\rm FM}+\tau_{\rm p}, \tag{4.2}$$

where $\tau_{\rm p}$ is the pulse duration. Using this time delay, the modulation of the trailing part of each pulse pair starts right after the leading pulse has left the phase modulator. In our setup $\tau_{\rm LPDc}=2\tau_{\rm FM}$, $\tau_{\rm p}=20$ ns. Hence $T_{\rm min}$ is given by

$$T_{\rm min}=\tau_{\rm AB}+20\,{\rm ns}, \tag{4.3}$$

where $\tau_{\rm AB}=T_{\rm total}/2-\tau_{\rm FM}$, furthermore the refractive index and length of the fiber, $n_{\rm fiber}=1.467$, $L_{\rm FM}$=0.5 m, $c=3\cdot 10^8$ m/s, so $\tau_{\rm FM}=n_{\rm fiber}L_{\rm FM}/c\simeq 2.45$ ns.

2) *Producing the actual modulation delay for Alice:* Using $T_{\rm min}$ here obtained from oscilloscope we don't account for the fact that Alice's card has a trigger-to-output delay of 238.5 sample clock cycles + 16 ns. But in the key generation phase directly the card is triggered to produce the phase modulation signal, so we still have to. With the 500 MSa/s base resolution setting applied in our tests and demonstration, this adds up to 493 ns, but in reality and together with the propagation times of the coaxial cables it has a value of 510 ns. (This value varies slightly, ostensibly due to the phase relation of the clocks in Alice's DAQ card and Bob's one, which provides the triggering optical signal.) Therefore, as part of this step we measure this total latency. We do so by setting an immediate output on triggering Alice's card and this time measuring the $\Delta T_{\rm cor}$ time difference of the triggering optical signal and the card's output (yielding a $-\Delta T_{\rm cor}$ correction to $T_{\rm min}$).

3) Tolerance of the delay: The light pulse train contains 480 pulse pairs after leaving Bob's side. We wait for the first pulse of the pair to pass PMa after returning from the FM. There is a 50 ns delay between the two pulses of each pair (due to the DL) and every pulse is 20 ns wide, furthermore, systematic uncertainties that are the 2 ns definition of our measurement and the 2 ns sampling time of the DAQ cards summed up for worst case estimation leave a $T_{\mathrm{margin}} = 26$ ns or to fit into with the start of our modulation signal. The cards have 16×2 ns resolution for trigger delay so the geometry of the components must be chosen so that integer times 32 ns is well between $T_{\mathrm{min}} - \Delta T_{\mathrm{cor}}$ and $T_{\mathrm{min}} - \Delta T_{\mathrm{cor}} + T_{\mathrm{margin}}$. Otherwise, we can tune ΔT_{cor} with the choice of coaxial cable, T_{margin} with switching to 625 MSa/s on the waveform generator or in extreme case T_{min} by setting τ_{FM} with a custom made fiber FM.

C. Setting up detection time series for Bob

In this last step we set up a time grid of 480 points with a frequency that we calculate from the detection events of a few thousand frames. What we want to be certain about then, is the arrival time of the first pulse, so that we can know the exact arrival times of every qubit in a frame. In this way we can later identify each and every detection resulting from sent qubits and can proceed with the basis reconciliation process to sift our key.

We aggregated the detection signal from 4500 frames in this step of the initialization. Groups of detection time tags gather around a set of time values compared to a signal that is synchronous with the laser firing (both are produced by the DAQ card controlling Bob). Some other time tags can also be observed between these groups that can be assessed as noise. Our task is to determine whether detection ticks belong to signal or noise.

1) Measurements: For every time tag in the aggregated data we add the distances from its neighbours. We then filter the majority of the noise ticks from the signals by comparing this sum to a threshold of 50 ns – some of the latter category can also be considered noise (typically near the edges of groups) in this phase. After this separation we only work with the signal data.

We calculate the mean value of time tags in each group. We continue by setting up multiple grids of 480 points with slightly different period times in the range 201 ns $> T_{\mathrm{grid}} >$ 199 ns and we search for the grid with minimal sum of absolute differences from the calculated mean values fixing the first grid point to the first mean value. We determine period time of the photon arrivals with 2 ps precision: for one channel $T_{\mathrm{grid,CH1}} = 199.996$ ns while for the other $T_{\mathrm{grid,CH2}} = 199.994$ ns so we initialized our data processing scripts with a common $T_{\mathrm{grid}} = 199.995$ ns. (The deviation from the expected 200 ns justifies opting to make this measurement, especially because this time grid is what helps us select signal and filter noise in the key generation phase.) The other required value was the first expected time of arrival, i. e. the mean value from the time tags in the first group on each channel. These values were $t_{1,\mathrm{CH1}} = 98964.089$ ns and $t_{1,\mathrm{CH2}} = 98974.460$ ns for SPAD1 and SPAD2 respectively, counted from emission. This 10 ns delay on channel 2 is due to the circulator that inserts 2 m of fiber before SPAD2.

2) Utility of results: To see why $T_{\mathrm{grid,CHi}}$ values can be determined in 10 ppm agreement and used to obtain T_{grid} with optimal accuracy, let's discuss how the aggregated frames are produced. Although Alice can perform no modulation throughout initialization, Bob alone can keep introducing a $\pi/2$ phase shift so that balanced optical power exits the interferometer arms. By doing so, for the same ordinal number of detection groups in the aggregated data of the 2 channels, signal photons originate from the same light pulse. Consequently, detection group mean values pairwise correspond to the same expected pulse arrival times, and the same holds for variances and higher moments, in addition the same time grid should give the best fit to them. We essentially have 2 measurements on 4500 points for the best fitting time grid which are sampled in identical circumstances and thus can be averaged to get a more precise value.

The choice of the first detection group mean as origin of the time grids to fit to data can be shown to be best as follows. The detector dead time being much greater than pulse width causes tick probability to decay exponentially over the first few pulses, resulting in a pronounced peak in tick number per group at the beginning of frames. The gradual 'opening' of the detectors makes this actually a global maximum for a frame, i. e. the first one the most sampled (most accurate) pulse arrival time.

Considering that this argument is based on the waiting time for the first impulse of the frame after the last one of the previous frame being much greater than the dead time, it makes the measurement also useful for estimating photon number per pulse. We can neglect the dead time and say that detectors tick any time they measure at least 1 photon in these pulses, detecting any photon with $\eta = 0.1 \ll 1$ efficiency. For such η the detection probability p can be calculated as if an N photon coherent pulse were being detected with η attenuation and an ideal detector [38], i. e.

$$p = 1 - \exp(-\eta N). \tag{4.4}$$

We have summed up the counts of the two detectors and divided it by the number of pulses 4500 sent. Then inverting the formula (4.4) we obtained $N = 0.62$ for the mean photon number in the pulses.

V. KEY DISTRIBUTION

This phase of the operation implement s interference-based QKD with the truth table of BB84. As prerequisite for the proper operation of this phase we have to run the initialization to configure our control scripts and components. The only variables that are set in this phase are the modulation bits and the physical signals for the modulators based on them. One thing left to determine in advance for this is the driving voltage set for the modulators to achieve the necessary phase shifts.

Earlier we measured the equivalent control voltage V_c for π phase shift on Bob's phase modulator having Alice's modulation off: then we get constructive interference at the SPAD2 output of Bob's 50 / 50 BS and desctructive at the SPAD1 output with Bob's PM off. The situation is interchanged if we introduce a phase shift of π, i. e. in both cases we get detection on only one SPAD neglecting noise. Modulating with 1.55 V random impacts are expected biased so that $N_{\mathrm{det,CH1}}/N_{\mathrm{det,CH2}} = Q = 0.859$ ($N_{\mathrm{det,CHi}}$ being the photon count on SPAD i) due to the attenuation of the circulator.

Alice needs 2 different bases, which means 4 different phase shifts altogether: 0, π, $\pi/2$ and $3\pi/2$. However, the current geometrical dimensions (τ_{FM} on Figure 2 in particular) constrain Alice to modulate pulses traversing the modulator in both directions, i. e. start to modulate as soon as a pulse *reaches* PMa *towards* FM so that the whole pulse is modulated through, albeit twice. Then pulses are orthogonally polarized on the two passes. The PM having polarization dependent characteristics such V_c values are sought that the sums of phase shifts in 2 directions are π, $\pi/2$ and $3\pi/2$ for the pulses. We performed an exhaustive search over a range of voltage values. At every inspected value we emitted 100 frames and calculated the $N_{\mathrm{det,CH1}}/N_{\mathrm{det,CH2}}$ quotient, comparing it to Q , this way we were able to determine 1.22 V for $\pi/2$.

When searching higher voltages for $3\pi/2$ we experienced high fluctuations and quotient values were far away from Q. Based on calculations we assumed that sinusoidal characteristics can be a close approximation for the operation of the PM but only when we modulate between $-\pi$ and π phases. Since the PM is operational at negative voltage s and our signal generator can output positive and negative voltages, the issue was solved by mirroring the 1.22 V for $\pi/2$ on zero point and search for the proper V_c for $3\pi/2$ there. With this idea we successfully found -1.23 V good enough for $3\pi/2$ phase shift. Our approach for finding π phase shift was based on the sinusoidal characteristics and two V_c values of 1.22 V for $\pi/2$ and an earlier estimated 3.56 V for $3\pi/2$ that we could not place in practice due to the unfavorable results mentioned before. Our assumption was that the demanded V_c for π should have been at the mean of this two values, that is 2.39 V. We checked this voltage with our scripts and so we got the presumed satisfying results. The control voltages paired to the bases in the implemented BB84 protocol applied in our system for the demonstration can be seen summarized in Table I.

TABLE I
Control Voltages for Bases

	phase [rad]	voltage [V]
Bob	0	0
	$\pi/2$	1.55
Alice	0	0
	π	2.39
	$\pi/2$	1.22
	$3\pi/2$	-1.23

The modulation delay in compliance with IV-B3, the DAQ card sampling frequency, memory and buffer size and trigger mode is set on Alice's side as well as parameters necessary for the waveform we would like to generate (e.g. analog / digital output, channel options, generation mode) by a Python module created to support the key distribution process. Exploiting the capabilities of the card we implemented the key distribution on Alice's side and we created the data processing at Bob to distill the key.

VI. Demonstration results

The truth table of BB84 regarding our system enumerates the 8 cases of basis pairing with Alice's and Bob's choices in Table II. We performed deterministic demonstration, i.e. we inspected all 8 cases by generating constant series for every initial key combination. This way tracing back the sources of errors in the results is as simple as comparing them to the truth table of the BB84 protocol.

For each case we prepared series of 100 frames. In the 4 matching basis choice scenarios we expected the sifted key bits at Bob to be the same as in Alice's initial key. In the data processing of the detected pulses we call an impact noise if it is outside a 25 ns radius centered at any expected arrival time in the grid we established in the initialization phase. Similarly, we call signal those impacts that are within this time interval. In matching basis choice scenarios we count the signal bits different from Alice's initial key values and name them false bits, the rest correct ones. From these data we can calculate the success rate of our implementation as the ratio of correct to all signal bits.

When dealing with different basis choices, we compare the bias photon count ratio to $Q = 0.859$ because for each such combination one detection tick means one sample of a modulation voltage measurement described in Section V. The percentage difference from this bias is a figure of merit for the accuracy of the modulation. The results of the demonstration are introduced in Tables III and IV.

In case of similar basis choices the results are introduced in Table III. These scenarios are responsible for producing sifted key bits. This means that the success rate calculated from this data is the determining factor in the quality of our demonstration. The last row of Table III shows that our implementation corresponds to the theoretical BB84 truth table in between 78.22 % and 97.49 % and that the more places in our system we apply modulation the more error we introduce into the signal detection. Still, the above 90 % values present promising first results.

The penultimate row of the Table IV shows that in a worst case scenario we have 5.33 % difference from Q, which occurs when we apply modulation on both sides. The first three columns indicate the error rate that a single modulation produces (4.89 % – 9.66 % that represents 14-30 false bits out of 100 frames). Together with previously mentioned results this suggests that the modulation creates interference that deviates from the optimal. Further adjustments of the modulation voltages may take care of this deviation, enhance success rate and fix Q_{measured} values more precisely to Q.

The cumulative results that have been calculated only from the same basis choice scenarios are summarized in Table V. Our system operates at 88.11 % success rate over physical layer based on 400 frames and altogether almost 2000 photon impacts. This only refers to the raw key bit success rate

TABLE II
BB84 TRUTH TABLE

Basis and phase choice of Alice	0 (0)	0 (0)	0 (π)	0 (π)	1 ($\pi/2$)	1 ($\pi/2$)	1 ($3\pi/2$)	1 ($3\pi/2$)
Alice bits ('encoding')	0	0	1	1	0	0	1	1
Basis choice of Bob	0 (0)	1 ($\pi/2$)	0 (0)	1 ($\pi/2$)	0 (0)	1 ($\pi/2$)	0 (0)	1 ($\pi/2$)
Bob bits ('measurement')	0	–	1	–	–	0	–	1

In practice, Alice and Bob sample an independent and uniformly distributed random variable on each transmission (within a frame) to select a combination. In 4 combinations they choose matching bases and in 4 different ones; in the former case their identical bits yield the sifted key bits and in the latter, marked with dashes (–), Bob collects data to be statistically tested against the hypothesis of the presence of an eavesdropper (Eve).

TABLE III
RESULTS FOR SAME BASIS CHOICE SCENARIOS

Phases (bases) (Bob – Alice)	0 - 0	0 - π	$\pi/2$ - $\pi/2$	$\pi/2$ - $3\pi/2$
Noise	25	34	46	53
Signal	439	423	464	482
False Detection (Out of Signal)	11	34	65	105
Success Rate [%]	**97.49**	**91.96**	**85.99**	**78.22**

TABLE IV
RESULTS FOR DIFFERENT BASIS CHOICE SCENARIOS

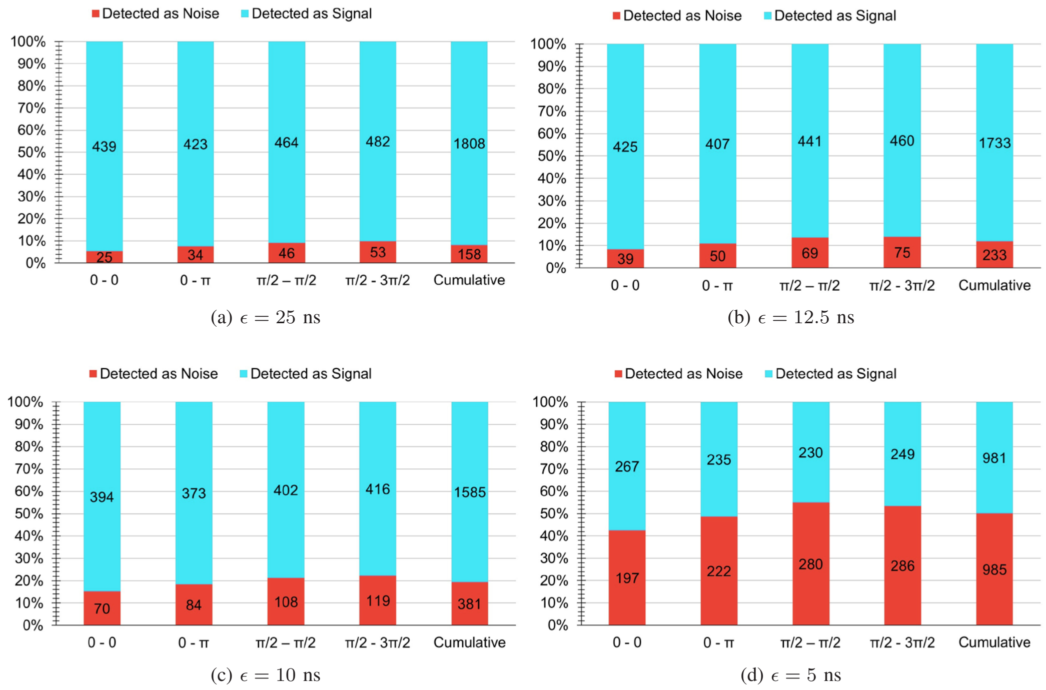
Phases (bases) (Bob – Alice)	0 - $\pi/2$	0 - $3\pi/2$	$\pi/2$ - 0	$\pi/2$ - π
Noise	46	44	53	49
Signal	671	659	679	653
Ticks on SPAD1 and on SPAD2 resp.	318 ; 353	288 ; 371	323 ; 356	283 ; 370
Q_{measured} $(N_{\text{det,CH1}}/N_{\text{det,CH2}})$	0.901	0.776	0.907	0.765
Difference from target $Q = 0.859$ [%]	**4.89**	**9.66**	**5.59**	**10.9**
Difference [detector ticks]	14.8	30.8	17.1	34.8

TABLE V
CUMULATIVE RESULTS FOR SAME BASIS CHOICE SCENARIOS

Noise	158
Signal	1808
False Detection (Out of Signal)	215
Success Rate [%]	**88.1084**

and does not concern the classical key distillation methods utilized by upper layers – our research only covers the physical layer of the system. Since this error rate of 11-12 % is only scratching the edge of the possibility to indicate the presence of an eventual eavesdropper [39], we've started to search for solutions and development directions to reach a success rate reliably above 90 %.

Besides the control voltages for the bases, the timing of the Alice-side modulation can also be a critical error source as well as our detection process. We modulate in two passes so we have to start the modulation earlier by 5 ns, i. e. two times the propagation time τ_{FM} between PMa and FM: T_{margin} is narrowed down to 21 ns. This is necessary for avoiding non-uniform modulation of the pulses. We may get less noisy transmission if we could synchronize to the instance



(a) $\epsilon = 25$ ns

(b) $\epsilon = 12.5$ ns

(c) $\epsilon = 10$ ns

(d) $\epsilon = 5$ ns

Fig. 3. Change in classification of ticks as signal or noise with varying ϵ

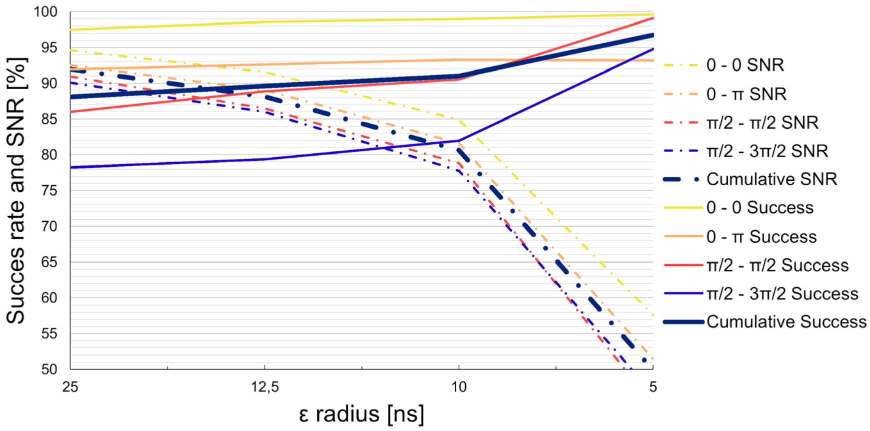


Fig. 4. Percentage SNRs and success rates as functions of ϵ in a finer sweep, noise seen to start dominating above cca. 10 ns

before the pulses arrive to the modulator back from the mirror but they've left PMa propagating towards it. This requires the extension of the fiber path between these components. The other critical part is the detection where we decided to consider the ticks within an $\epsilon = 25$ ns radius around every point in the grid. After getting the demonstration results we considered modifying this radius in the hope of achieving better success rates. Our motivation was to filter more noise from our transmitted pulses so we started to shrink this 50 ns time interval – first to 25 ns, then 20 ns and finally to 10 ns ($\epsilon = 12.5, 10, 5$ ns, respectively). This way we also exclude more signal bits, decreasing our SNR that can be seen on Figure 3. The question is how this software modification will affect our success rates. Recalculating the demonstration results with these new ϵ values we arrive at Figure 4. We can observe that the curves representing the success rates rise monotonously with the decrease of the radius. Setting $\epsilon = 5$ ns the curve representing the cumulative success rate can reach 96.73 % – meanwhile the SNR, unfortunately drops dramatically. This means that we can make our protocol 8 % more effective with bit identification, making it more secure but signal bit rates will fall resulting in slower key generation. We need to find the optimal ϵ, which depends on our optimizing strategy. We calculate trends in success rate and SNR on the 3 sections. Between 5 ns and 10 ns the slope of the SNR significantly drops so we can choose 10 ns to be a threshold point as one strategy that results in over 90 % success rate and only a 10 % reduction in SNR. However, a preferable solution is to improve the SNR of the system in advance so that our corresponding curves in this diagram could shift towards smaller ϵ, while the ones representing success rates to wards greater. This way we could improve the efficiency of our system to a great extent only slightly reducing bit generation rate.

VII. OUTLOOK

We are currently at the start of a next stage with various kinds of research goals to be set.

- *Optimization of current components*
  Adjustments of PM voltages with reference to Table IV, in particular $\Delta Q = Q_{\text{measured}} - Q$ – these results are significantly higher than what we expected based on previous results from determining these voltages.
  We can also try using gated detection as in the original paper proposing this architecture. It's worth pointing out, though, that the time grid fitting method for detection is the mechanism that enables our implementation to work with detectors in (or potentially only capable of) free running mode.
- *Hardware improvement*
  Our EVOAs have the feature of modulation, which gives the idea to set higher attenuation values when Alice isn't transmitting or Bob's detectors aren't expecting frames. Yet with the bandwidth of currently used MEMS devices is insufficient for this.
- *Improvement towards field applicability*
  Small form factor pluggable (SFP) lasers have become more and more commonly available and widespread for use in telecommunication, which makes it desirable to test performance with such a source. Furthermore, the oscilloscope would have to be substituted with the similarly common time to digital conversion (TDC).
  A step towards field practicality would be fully automating DAQ cards to operate independently from PCs (like they do now).

VIII. CONCLUSION

In this paper we made a successful approach to realize quantum key distribution in an optical fiber system, for which a precise modulation timing was implemented at Alice. The system structure required synchronisation between Alice and Bob taking emitted pulse frames as reference. Our calculations revealed a tight criteria for the time window targeted with the modulation signal. For establishing this timing mechanism for key distribution, we introduced an initial phase within necessary adjustments of optical power levels and optical length measurements within 2 ns accuracy are carried out. In subsections IV-A and IV-B we describe our solutions for adjusting our system parameters to find the critical modulation time window.

We implemented BB84 protocol in our system setting up the bases for modulation on both sides as described in section V by performing exhaustive search to find the control voltages responsible for the appropriate phase shifts on the laser pulses. A detection grid was also set up at Bob during initialization phase introduced in subsection IV-C. The grid consisting of the expected photon arrival time series includes a basic initial noise filtering. Furthermore, the adjustment of a single parameter of noise filtering gives the opportunity of setting efficiency adjustments in raw key quality and generation by software.

Based on the first results our system operates at 88.11 % success rate for the aggregated data. Part of the deterministic tests we achieved 97.49 % as the best individual performance among base pairings. The demonstration showed the potential in our system to reach this level also in overall efficiency with parameter optimization, hardware improvements and more efficient software techniques. We intend to implement these upgrades by the principles of plug & play that we have followed during our previous efforts, as well.

Our work contributes to the research goal of providing quantum security in commercial communication networks that stands economically.

REFERENCES

[1] S. Imre, "Quantum computing and communications - introduction and challenges," *Comput. Electr. Eng.*, vol. 40, no. 1, p. 134–141, Jan. 2014, DOI: 10.1016/j.compeleceng.2013.10.008.

[2] ——, "Quantum communications: explained for communication engi- neers," *IEEE Communications Magazine*, vol. 51, no. 8, pp. 28–35, August 2013, DOI: 10.1109/MCOM.2013.6576335.

[3] C. H. Bennett and G. Brassard, "Quantum cryptography: Public key distribution and coin tossing," *Theoretical Computer Science*, vol. 560, p. 7–11, Dec 2014, DOI: 10.1016/j.tcs.2014.05.025.

[4] L. B. Laszlo Gyongyosi and S. Imre, "A Survey on Quantum Key Distribution," *Infocommunications Journal*, vol. XI, no. 2, pp. 14 – 21, 6 2020, DOI: 10.36244/ICJ.2019.2.2.

[5] S. Imre and F. Balázs, *Quantum Computing Basics*. John Wiley & Sons, Ltd, 2004, ch. 2, pp. 7–42, DOI: 10.1002/9780470869048.ch2.

[6] D. Kobor and E. Udvary, "Optimisation of Optical Network for Continuous-Variable Quantum Key Distribution by Means of Simulation," *Infocommunications Journal*, vol. XII, no. 2, pp. 18 – 24, 6 2020, DOI: 10.36244/ICJ.2020.2.3.

[7] A. Mraz, Z. Kis, S. Imre, L. Gyongyosi, and L. Bacsardi, "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*, vol. 45, 04 2017, DOI: 10.1002/cta.2347.

[8] Ágoston Schranz and E. Udvary, "Mathematical analysis of a quantum random number generator based on the time difference between photon detections," *Optical Engineering*, vol. 59, no. 4, pp. 1 – 13, 2020, DOI: 10.1117/1.OE.59.4.044104.

[9] Y. Mao, B.-X. Wang, C. Zhao, G. Wang, R. Wang, H. Wang, F. Zhou, J. Nie, Q. Chen, Y. Zhao, Q. Zhang, J. Zhang, T.-Y. Chen, and J.-W. Pan, "Integrating quantum key distribution with classical communications in backbone fiber network," *Opt. Express*, vol. 26, no. 5, pp. 6010–6020, Mar 2018, DOI: 10.1364/OE.26.006010.

[10] M. Gündoğan, J. S. Sidhu, V. Henderson, L. Mazzarella, J. Wolters, D. K. L. Oi, and M. Krutzik, "Proposal for space-borne quantum memories for global quantum networking," *npj Quantum Information*, vol. 7, no. 1, p. 128, Aug 2021, DOI: 10.1038/s41534-021-00460-9.

[11] R. Kaltenbaek, A. Acin, L. Bacsardi, P. Bianco, P. Bouyer, E. Diamanti, C. Marquardt, Y. Omar, V. Pruneri, E. Rasel, B. Sang, S. Seidel, H. Ulbricht, R. Ursin, P. Villoresi, M. van den Bossche, W. von Klitzing, H. Zbinden, M. Paternostro, and A. Bassi, "Quantum technologies in space," *Experimental Astronomy*, Jun 2021, DOI: 10.1007/s10686-021-09731-x.

[12] M. Mastriani, S. Iyengar, and L. Kumar, "Satellite quantum communication protocol regardless of the weather," *Optical and Quantum Electronics*, vol. 53, 04 2021, DOI: 10.1007/s11082-021-02829-8.

[13] L. Bacsardi, "On the way to quantum-based satellite communication," *Communications Magazine*, IEEE, vol. 51, pp. 50–55, 08 2013, DOI: 10.1109/MCOM.2013.6576338.

[14] J. S. Sidhu, S. K. Joshi, M. Gündoğan, T. Brougham, D. Lowndes, L. Mazzarella, M. Krutzik, S. Mohapatra, D. Dequal, G. Vallone, P. Villoresi, A. Ling, T. Jennewein, M. Mohageg, J. G. Rarity, I. Fuentes, S. Pirandola, and D. K. L. Oi, "Advances in space quantum communications," *IET Quantum Communication*, vol. n/a, no. n/a, Jul 2021, DOI: 10.1049/qtc2.12015.

[15] K. Günthner, I. Khan, D. Elser, B. Stiller, Ömer Bayraktar, C. R. Müller, K. Saucke, D. Tröndle, F. Heine, S. Seel, P. Greulich, H. Zech, B. Gütlich, S. Philipp-May, C. Marquardt, and G. Leuchs, "Quantum-limited measurements of optical signals from a geostationary satellite," *Optica*, vol. 4, no. 6, pp. 611–616, Jun 2017, DOI: 10.1364/OPTICA.4.000611.

[16] C. Simon, "Towards a global quantum network," *Nature Photonics*, vol. 11, no. 11, pp. 678–680, Nov 2017, DOI: 10.1038/s41566-017-0032-0.

[17] P. Villoresi, T. Jennewein, F. Tamburini, M. Aspelmeyer, C. Bonato, R. Ursin, C. Pernechele, V. Luceri, G. Bianco, A. Zeilinger, and et al., "Experimental verification of the feasibility of a quantum channel between space and earth," *New Journal of Physics*, vol. 10, no. 3, p. 033038, Mar 2008, DOI: 10.1088/1367-2630/10/3/033038.

[18] L. Calderaro, C. Agnesi, D. Dequal, F. Vedovato, M. Schiavon, A. Santamato, V. Luceri, G. Bianco, G. Vallone, and P. Villoresi, "Towards quantum communication from global navigation satellite system," *Quantum Science and Technology*, vol. 4, no. 1, p. 015012, dec 2018, DOI: 10.1088/2058-9565/aaefd4.

[19] R. Ursin, F. Tiefenbacher, T. Schmitt-Manderbach, H. Weier, T. Scheidl, M. Lindenthal, B. Blauensteiner, T. Jennewein, J. Perdigues, P. Trojek, B. Ömer, M. Fürst, M. Meyenburg, J. Rarity, Z. Sodnik, C. Barbieri, H. Weinfurter, and A. Zeilinger, "Entanglement-based quantum com- munication over 144km," *Nature Physics*, vol. 3, no. 7, pp. 481–486, Jul 2007, DOI: 10.1038/nphys629.

[20] K. Resch, M. Lindenthal, B. Blauensteiner, H. Böhm, A. Fedrizzi, C. Kurtsiefer, A. Poppe, T. Schmitt-Manderbach, M. Taraba, R. Ursin, P. Walther, H. Weier, H. Weinfurter, and A. Zeilinger, "Distributing en- tanglement and single photons through an intra-city, free-space quantum channel," *Opt. Express*, vol. 13, no. 1, pp. 202–209, Jan 2005, DOI: 10.1364/OPEX.13.000202.

[21] F. Steinlechner, S. Ecker, M. Fink, B. Liu, J. Bavaresco, M. Huber, T. Scheidl, and R. Ursin, "Distribution of high-dimensional entanglement via an intra-city free-space link," *Nature Communications*, vol. 8, no. 1, p. 15971, Jul 2017, DOI: 10.1038/ncomms15971.

[22] R. Valivarthi, M. G. Puigibert, Q. Zhou, G. H. Aguilar, V. B. Verma, F. Marsili, M. D. Shaw, S. W. Nam, D. Oblak, and W. Tittel, "Quantum teleportation across a metropolitan fibre network," *Nature Photonics*, vol. 10, no. 10, pp. 676–680, Oct 2016, DOI: 10.1038/nphoton.2016.180.

[23] Q.-C. Sun, Y.-L. Mao, S.-J. Chen, W. Zhang, Y.-F. Jiang, Y.-B. Zhang, W.-J. Zhang, S. Miki, T. Yamashita, H. Terai, X. Jiang, T.-Y. Chen, L.-X. You, X.-F. Chen, Z. Wang, J.-Y. Fan, Q. Zhang, and J.-W. Pan, "Quantum teleportation with independent sources and prior entanglement distribution over a network," *Nature Photonics*, vol. 10, no. 10, pp. 671–675, Oct 2016, DOI: 10.1038/nphoton.2016.179.

[24] M. Mastriani, S. S. Iyengar, and K. J. Latesh Kumar, "Bidirectional teleportation for underwater quantum communications," *Quantum Information Processing*, vol. 20, no. 1, p. 22, Jan 2021, DOI: 10.1007/s11128-020-02970-5.

[25] C. Elliott, A. Colvin, D. Pearson, O. Pikalo, J. Schlafer, and H. Yeh, "Current status of the DARPA quantum network," in *Quantum Information and Computation* III, E. J. Donkor, A. R. Pirich, and H. E. Brandt, Eds., vol. 5815, International Society for Optics and Photonics. SPIE, 2005, pp. 138 – 149, DOI: 10.1117/12.606489.

[26] M. Peev, C. Pacher, R. Alléaume, C. Barreiro, J. Bouda, W. Boxleitner, T. Debuisschert, E. Diamanti, M. Dianati, J. F. Dynes, S. Fasel, S. Fossier, M. Fürst, J.-D. Gautier, O. Gay, N. Gisin, P. Grangier, A. Happe, Y. Hasani, M. Hentschel, H. Hübel, G. Humer, T. Länger, M. Legré, R. Lieger, J. Lodewyck, T. Lorünser, N. Lütkenhaus, A. Marhold, T. Matyus, O. Maurhart, L. Monat, S. Nauerth, J.-B. Page, A. Poppe, E. Querasser, G. Ribordy, S. Robyr, L. Salvail, A. W. Sharpe, A. J. Shields, D. Stucki, M. Suda, C. Tamas, T. Themel, R. T. Thew, Y. Thoma, A. Treiber, P. Trinkler, R. Tualle-Brouri, F. Vannel, N. Walenta, H. Weier, H. Weinfurter, I. Wimberger, Z. L. Yuan, H. Zbinden, and A. Zeilinger, "The SECOQC quantum key distribution network in vienna," *New Journal of Physics*, vol. 11, no. 7, p. 075001, jul 2009, DOI: 10.1088/1367-2630/11/7/075001.

[27] D. Stucki, M. Legré, F. Buntschu, B. Clausen, N. Felber, N. Gisin, L. Henzen, P. Junod, G. Litzistorf, P. Monbaron, and et al., "Long-term performance of the swissquantum quantum key distribution network in a field environment," *New Journal of Physics*, vol. 13, no. 12, p. 123001, Dec 2011, DOI: 10.1088/1367-2630/13/12/123001.

[28] M. Sasaki, M. Fujiwara, H. Ishizuka, W. Klaus, K. Wakui, M. Takeoka, S. Miki, T. Yamashita, Z. Wang, A. Tanaka, K. Yoshino, Y. Nambu, S. Takahashi, A. Tajima, A. Tomita, T. Domeki, T. Hasegawa, Y. Sakai, H. Kobayashi, T. Asai, K. Shimizu, T. Tokura, T. Tsurumaru, M. Matsui, T. Honjo, K. Tamaki, H. Takesue, Y. Tokura, J. F. Dynes, A. R. Dixon, A. W. Sharpe, Z. L. Yuan, A. J. Shields, S. Uchikoga, M. Legré, S. Robyr, P. Trinkler, L. Monat, J.-B. Page, G. Ribordy, A. Poppe, A. Allacher, O. Maurhart, T. Länger, M. Peev, and A. Zeilinger, "Field test of quantum key distribution in the tokyo qkd network," *Opt. Express*, vol. 19, no. 11, pp. 10 387–10 409, May 2011, DOI: 10.1364/OE.19.010387.

[29] S.-K. Liao, W.-Q. Cai, J. Handsteiner, B. Liu, J. Yin, L. Zhang, D. Rauch, M. Fink, J.-G. Ren, W.-Y. Liu, and et al., "Satellite-relayed intercontinental quantum network," *Physical Review Letters*, vol. 120, no. 3, Jan 2018, DOI: 10.1103/physrevlett.120.030501.

[30] Y.-A. Chen, Q. Zhang, T.-Y. Chen, W.-Q. Cai, S.-K. Liao, J. Zhang, K. Chen, J. Yin, J.-G. Ren, Z. Chen, S.-L. Han, Q. Yu, K. Liang, F. Zhou, X. Yuan, M.-S. Zhao, T.-Y. Wang, X. Jiang, L. Zhang, W.-Y. Liu, Y. Li, Q. Shen, Y. Cao, C.-Y. Lu, R. Shu, J.-Y. Wang, L. Li, N.-L. Liu, F. Xu, X.-B. Wang, C.-Z. Peng, and J.-W. Pan, "An integrated space-to-ground quantum communication network over 4,600 kilometres," *Nature*, vol. 589, no. 7841, pp. 214–219, Jan 2021, DOI: 10.1038/s41586-020-03093-8.

[31] S. E. Ltd, "Cryptography secures swiss elections," (accessed: 03.08.2021). [Online]. Available: https://optics.org/article/31646

[32] R. Naik and P. Reddy, "Towards secure quantum key distribution protocol for wireless lans: a hybrid approach," *Quantum Information Processing*, vol. 14, 12 2015, DOI: 10.1007/s11128-015-1129-3.

[33] "Sk telecom continues to protect its 5g network with quantum cryptography technologies," Mar 2019, (accessed: 03.08.2021). [Online]. Available: https://www.idquantique.com/sk-telecom-continues-to-protect-its-5g-network-with-quantum-cryptography-technologies/

[34] R. Asif and W. J. Buchanan, "Recent progress in the quantum-to-the-home networks," in *Telecommunication Networks*, M. A. Matin, Ed. Rijeka: IntechOpen, 2019, ch. 2, DOI: 10.5772/intechopen.80396.

[35] "Banking," Aug 2020, (accessed: 03.08.2021). [Online]. Available: https://www.idquantique.com/random-number-generation/applications/banking/

[36] A. M. Lewis and M. Travagnin, "A secure quantum communications infrastructure for europe," Joint Research Centre, Report JRC116937, 2019.

[37] D. Stucki, N. Gisin, O. Guinnard, G. Ribordy, and H. Zbinden, "Quan- tum key distribution over 67 km with a plug&play system," *New Journal of Physics*, vol. 4, pp. 41–41, jul 2002, DOI: 10.1088/1367-2630/4/1/341.

[38] G. Zambra, A. Andreoni, M. Bondani, M. Gramegna, M. Genovese, G. Brida, A. Rossi, and M. G. A. Paris, "Experimental reconstruction of photon statistics without photon counting," *Phys. Rev. Lett.*, vol. 95, p. 063602, Aug 2005, DOI: 10.1103/PhysRevLett.95.063602.

[39] P. W. Shorand J.Preskill,"Simple proof of security of the bb84 quantum key distribution protocol," *Phys. Rev. Lett.*, vol. 85, pp. 441–444, Jul 2000, DOI: 10.1103/PhysRevLett.85.441.

Márton Czermann received his BSc degree in 2020 in Electrical Engineering from the Budapest University of Technology and Economics. He started his MSc studies at the Department of Networked Systems and Services at the same faculty. He joined a fiber-based DV-QKD project in 2018 which realizes the BB84 algorithm and a free space entanglement-based DV-QKD project in 2019. Since 2020, he is also a participant of Quantum Future Academy, Berlin.

Péter Trócsányi MSc student with the major of Research Physicist. In 2019 started investigating polarization phenomena and joined Ericsson Hungary RD as an intern. He received his BSc in Electrical Engineering from the Budapest University of Technology and Economics in 2020 prototyping optical quantum random number generators for his thesis. That year he started his Physics studies which involved shifting his focus from fiber to free space optics. He joined Wigner Research Center for Physics as an intern in 2021. He has been member of Simonyi (since 2016) and Wigner (since 2020) Colleges for Advanced Study.

Zsolt Kis gained the PhD degree in physics in 2000 at the University of Szeged. His main research field is quantum optics. Recently he has been leading a research group on developing single photon sources in the visible and telecommunication wavelength domain. He has participated in the development of the first Hungarian CV QKD system. Now he leads the development of a new CV QKD system and a DV QKD system which realizes the BB84 algorithm.

Benedek Kovács edge computing expert. The main responsibilities are engineering the evolution of telecommunication networks in the area of 5G and edge computing and managing innovation projects and university collaborations. Joined Ericsson in 2005 as a software developer and tester, and later worked as a system engineer, served as the characteristics, performance management and reliability specialist in the development of the 4G VoLTE solution, since then he focuses on edge computing. Kovács holds an MSc in information engineering and a PhD in mathematics from the Budapest University of Technology and Economics.

László Bacsárdi (M'07) received his MSc degree in 2006 in Computer Engineering from the Budapest University of Technology and Economics (BME) and his PhD in 2012. He is corresponding member of the International Academy of Astronautics (IAA). Between 2009 and 2020, he worked at the University of Sopron, Hungary in various positions including Head of Institute of Informatics and Economics. Since 2020, he is associate professor at the Department of Networked Systems and Services, BME and head of Mobile Communications and Quantum Technologies Laboratory. His current research interests are quantum computing, quantum communications and ICT solutions developed for Industry 4.0. He is the past chair of the Telecommunications Chapter of the Hungarian Scientific Association for Infocommunications (HTE), Vice President of the Hungarian Astronautical Society (MANT). Furthermore, he is member of AIAA, IEEE and HTE as well as alumni member of the UN established Space Generation Advisory Council (SGAC). In 2017, he won the IAF Young Space Leadership Award from the International Astronautical Federation.

Test generation algorithm for the All-Transition-State criteria of Finite State Machines

Gábor Árpád Németh and Máté István Lugosi

***Abstract*—In the current article a novel test generation algorithm is presented for deterministic finite state machine specifications based on the recently introduced All-Transition-State criteria. The size of the resulting test suite and the time required for test suite generation are investigated through analytical and practical analyses and are also compared to the Transition Tour, Harmonized State Identifiers and random walk test generation methods. The fault detection capabilities of the different approaches are also investigated with simulations applying randomly injected transfer faults.**

***Index Terms*—model-based testing, conformance testing, finite state machine, test generation algorithms**

I. Introduction

Testing plays a vital role in the software development life cycle. The complexity of software is continuously increasing, whereas nowadays the time frame between two releases becomes shorter, raising the probability of faults. Compared to the complexity of the problem, only limited resources are allocated for testing to provide adequate quality for the end product. Although the execution of test cases are automated in most big software companies, test design is typically still done manually, which is a very time consuming process. To cope with this challenge, one can raise the level of automation for the design of test cases as well. If the requirements of the product are described in a formal model specification, then the test cases can be generated automatically from this model to fulfill given testing goals. This area of testing is called model-based testing (MBT).

Several formal models exist for system specifications, such as behaviour trees [8], Finite State Machines (FSMs) [6], [15], [18] and labelled transition systems [6]. This article focuses on FSM formal models, which have been extensively used in diverse areas such as telecommunication software and protocols [13], [14], software related to lexical analysis and pattern matching [3], hardware design [24] and embedded systems [5].

In this article we present a novel test generation algorithm for finite state machine specifications. Our approach is based on the All-Transition-State (ATS) criteria introduced in [10] and uses elements of the Chinese Postman Tour algorithms [9].

The body of the article is organized as follows. Section II discusses related terms regarding graphs, FSMs and conformance testing. The most relevant FSM-based test generation algorithms that are used as a reference point when evaluating our algorithm are also discussed here. Section III introduces our new test generation algorithm for the All-Transition-State criteria, demonstrates it through an example and provides an analysis of its complexity. Section IV presents simulations investigating the test generation time, the overall length of the test sequences and the fault coverage of our algorithm compared to existing methods. The main results of the paper are concluded in Section V with possible future directions.

II. Preliminaries

A. Graphs

A *directed graph* is a $G = (V, E)$ (possibly with loop and parallel arcs), where $V = \{s_1, \ldots, s_n\}$ denotes the set of *nodes* and $E = \{e_1, \ldots, e_m\}$ denotes the set of *ordered* pairs of nodes (s_k, s_l) called *directed edges* or *arcs*. In a *weighted directed graph* a number – called *weight* – is assigned to each arc.

A *directed walk* is a finite and alternative sequence of nodes and arcs, $(s_1, e_1, s_2, ...e_{n-1}, s_n)$, where each s_k, s_{k+1} consecutive nodes are the end points of an intermediate edge e_k. A *directed trail* is a directed walk in which all arcs are distinct, a *directed path* is a directed trail in which all nodes are distinct.

A *directed cycle* is a directed trail where the first node is the same as the last node of the sequence. A directed graph is acyclic if it does not contain any directed cycles. A *spanning forest* of a G is an acyclic subgraph of G. A *spanning tree* ST of G is an acyclic subgraph of G which includes all of the nodes of G and exactly $|V| - 1$ arcs which are directed away from root node s_0, so that there is exactly one path from s_0 to any other node. An *inverse spanning tree* TS of G is an acyclic subgraph of G which includes all of the nodes of G and exactly $|V| - 1$ arcs which are directed toward a root node s_0, such a way that from every node $s_k \in V$ there exists exactly one directed path to s_0.

A directed graph is *strongly connected* if there exists a directed path between any two given nodes. The *strongly connected components* (SCCs) of graph G are the maximal strongly connected subgraphs of graph G.

Let the number of arcs originating from node s_j be denoted by $deg^+(s_j)$ (outdegree), and the number of arcs that lead to node s_j by $deg^-(s_j)$ (indegree). We say that node s_j is *balanced* iff $deg^-(s_j) = deg^+(s_j)$, otherwise unbalanced. We say that a directed graph is *Eulerian*, if it is strongly connected and balanced for every node.

A *bipartite graph* $G_B = (V^-, V^+, E)$ is a graph whose nodes can be divided into two disjoint and non-empty sets

Gábor Árpád Németh and Máté István Lugosi are with the Department of Computer Algebra, Faculty of Informatics, Eötvös Loránd University, H-1117 Budapest, Pázmány Péter sétány 1/C., Hungary, e-mail: nga@inf.elte.hu, mate.lugosi@gmail.com

DOI: 10.36244/ICJ.2021.3.6

denoted with V^- and V^+ and every edge in E connects a node in V^- to one in V^+. A *matching* in G_B is a $E_m \subseteq E$ subset of its edges, where none of them share the same node. If sets V^- and V^+ cover the same number of nodes, then a *minimum weighted perfect matching* of G_B may exist, that covers all nodes of sets V^- and V^+ and the overall weights of its edges are minimal.

B. Finite State Machines

A Mealy Finite State Machine (abbreviated as 'FSM' in the rest of the article) M is a quadruple $M = (I, O, S, T)$ where I, O, and S are the finite and non-empty sets of *input symbols*, *output symbols* and *states*, respectively. T is the finite and non-empty set of *transitions* between states. Each transition $t \in T$ is a quadruple $t = (s_j, i, o, s_k)$, where $s_j \in S$ is the start state, $i \in I$ is an input symbol, $o \in O$ is an output symbol and $s_k \in S$ is the next state. The number of states, inputs and transitions of an FSM are denoted by $n = |S|$, $p = |I|$ and $m = |T|$, respectively.

An FSM can be represented with a *state transition graph*, which is a directed labelled graph whose nodes and arcs correspond to the states and transitions, respectively. Each arc is labeled with the input and the output, written as i/o, associated with the transition.

FSM M is *deterministic*, if for each (s_j, i) state-input pair there exists at most one transition in T, otherwise it is *non-deterministic*. If there is at least one transition $t \in T$ for all state-input pairs, the machine is said to be *completely specified*, otherwise it is *partially specified*.

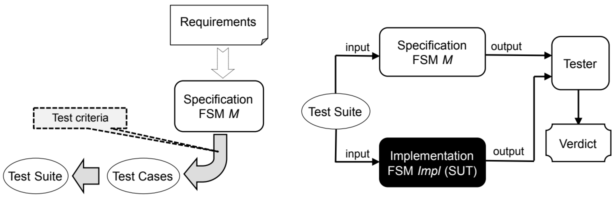
In case of deterministic FSMs the output and the next state of a transition can be given as a function of the start state and the input of a transition, where λ: $S \times I \rightarrow O$ denotes the *output function* and δ: $S \times I \rightarrow S$ denotes the *next state function*. Let us extend δ and λ from input symbols to finite *input sequences* I^* as follows: for a state s_1, an input sequence $x = i_1, \ldots, i_k$ takes the machine successively to states $s_{j+1} = \delta(s_j, i_j)$, $j = 1, \ldots, k$ with the final state $\delta(s_1, x) = s_{k+1}$, and produces an *output sequence* $\lambda(s_1, x) = o_1, \ldots, o_k$, where $o_j = \lambda(s_j, i_j)$, $j = 1, \ldots, k$. The string concatenation operator is denoted by ".".

Two states, s_j and s_l of FSM M are *distinguishable*, iff there exists an $x \in I^*$ input sequence – called a *separating sequence* – that produces different output for these states, i.e.: $\lambda(s_j, x) \neq \lambda(s_l, x)$. Otherwise states s_j and s_l are *equivalent*. A machine is *reduced*, if no two states are equivalent.

An FSM M has a *reset message*, if there exists a special input symbol $r \in I$ that takes the machine from any state back to the s_0 initial state: $\exists r \in I : \forall s_j : \delta(s_j, r) = s_0$. The *reset is reliable* if it is guaranteed to work properly in any implementation machine $Impl$ of M.

C. Conformance testing

The structure of FSM model-based test generation is shown in Figure 1(a): A formal specification model denoted by FSM M is derived from the requirements. From FSM M – according to some preset test criteria – *test cases* can be automatically generated; these are the pairs of input sequences and expected output sequences of M. A set of test cases form a *test suite*. This test suite then can be applied to the System Under Test (SUT) that can be considered as an $Impl$ implementation machine of specification M – see Figure 1(b). Note that machine $Impl$ can be considered as a black box with unknown internal structure, one can only observe its output responses upon a given input sequence. The role of *conformance testing* is to check if the observed output sequences of $Impl$ are equivalent to the expected results derived from M – i.e. to check if $Impl$ *conforms* to M.



(a) Test generation (b) Conformance testing

Figure 1. Model-based testing

D. FSM Fault Models

FSM fault models describe the assumptions of the test engineer about implementation machine $Impl$ as SUT. A usual approach is that the faults do not increase the number of the states specified in FSM M [15], thus the fault model of [7] and [4] are typically restricted to the following two types of faults [15]:

I. Output fault: for a given state-input pair FSM $Impl$ produces an output that is different from the one that is specified in FSM M.
II. Transfer fault: for a given state-input pair FSM $Impl$ goes into a state that differs from the state specified in FSM M.

E. Test generation methods

In the following we discuss relevant FSM-based test generation methods that are used as reference points when comparing the performance of our new algorithm. Note that the Transition Tour is discussed in more detail because its elements are reused in our method.

1) Random walk: Starting from the initial state, in each step a transition leading from the current state is chosen randomly and traversed entering a new state until a given stop condition is fulfilled. Various stop conditions – such as a percentage of input/output symbols, visited states or transitions – can be selected based on testing goals.

Although this approach can be useful for exploratory testing, it is impractical for the functional testing of a large-scale software as the length of the test sequence can be much longer than the optimal solution.

2) Harmonized State Identifiers: The Harmonized State Identifiers (HSI) [17], [25] state verification method can be used to create a structured test suite for reduced, deterministic, strongly connected FSMs with reliable reset capability [28]. The resulting algorithm is the generalization of the W [7] and Wp [11] methods and it guarantees to discover all output and transfer faults of FSM $Impl$. According to simulations of [27] this is the most efficient of the W/Wp/HSI triple.

Each test case of HSI consists of the following parts:

- A *state cover set* $Q = \{q_1, \ldots, q_n\}$ responsible for reaching all states; the problem can be reduced to creating a spanning tree ST from initial state s_0.
- A *separating family of sequences* of Z responsible for verifying end states. The Z set is a collection of sets $Z_i, i = 1, \ldots, n$ of sequences (one set for each state) where for every non-identical pair of states s_i, s_j there exists a separating sequence. The Z set can be represented with a spanning forest over a state pair graph, the arcs of which are directed to state pairs that have a separating input [23].

Based on the parts discussed above, the algorithm consists of two stages, one responsible for identifying all states of the machine and the other for checking all remaining transitions.

The resulting test suite consists of no more than $p \cdot n^2$ test sequences, each one with a length less than $2 \cdot n$ interposed with the reset symbol [28]. Thus, the total length of the resulting test suite and the complexity of test generation is $O(p \cdot n^3)$.

3) Transition Tour: The Transition Tour (TT) [19] algorithm produces a test sequence that visits every transition of a reduced, deterministic, strongly connected specification FSM M at least once and returns to the initial state. This is the shortest tour that provides 100% state- and transition coverage of the specification. It guarantees to discover all output faults, but does not guarantee to find transfer faults.

The problem of generating the TT test sequence can be reduced to the Directed Chinese Postman Problem (DCPP) [9] with unit costs for the arcs of graph G (where G corresponds to FSM M). There are multiple algorithms [9], [16], [22] that solve this problem, typically consisting of two major parts:

I. Augmenting the original graph G by duplicating some arcs to make it Eulerian.
II. Finding an Euler tour over an Eulerian graph G_E.

I. Augmenting the original graph to make it Eulerian: Since the goal is to generate the shortest possible test sequence, minimal additional arcs should be added. This is achieved by finding a *minimum weighted perfect matching* on a *bipartite graph* G_B, with group V^- representing the $deg^+ < deg^-$ *negative balanced* nodes and group V^+ representing the $deg^+ > deg^-$ *positive balanced* ones. The weight on edges represents the shortest directed paths from V^- to V^+ nodes of the original graph G in the following way: As the arcs of G are considered to have unit costs, the lengths of the shortest paths are measured as the number of arcs they contain. Every *unbalanced* node s_k of graph G is represented by $|balance(s_k)| = |deg^+(s_k) - deg^-(s_k)|$ number of nodes in bipartite graph G_B ensuring that the matching will contain exactly $|balance(s_k)|$ number of arcs incident to s_k. Once this is done, the graph G can be augmented into an Eulerian graph G_E by multiplicating the arcs along the paths represented by the edges in G_B used in the matching.

II. Finding an Euler tour: After graph G has been augmented to an Eulerian graph G_E, an Euler tour in G_E can be found. There are two ways to represent Euler-tours: the edge-pairing representation and the next-node representation [9]. Here the latter one is considered, which defines an order of the outgoing arcs for each node and in the l_{th} visit of the node, the l_{th} arc is used in this ordering. Finding an Euler tour in this case can be reduced to the creation of an inverse spanning tree TS [9]. To find an Euler tour over G_E one can start at the initial node and specify an arbitrary order for outgoing arcs from each node with the restriction that the arc that is in TS will be the last in the ordering.

The presented method that produces a TT test sequence has $O(n^3 + m)$ time complexity. The lengths of the resulting test sequences is $O(m)$.

III. All-Transition-State algorithm

We have created a novel heuristic test generation algorithm for reduced, deterministic, strongly connected FSM models based on the All-Transition-State (ATS) criteria introduced in [10]. Note that the original criteria formulates three formal conditions that the test suite should satisfy, but since the third one is applicable for the guarding condition of the EFSM (Extended Finite State Machine) models, we focus on the following two:

I. For all t transitions: The test suite should cover at least one walk that contains t and then reaches all states of FSM M.
II. There has to be at least one walk to all states which does not include transition t (if feasible).

The motivation behind applying these conditions are the following: (1) Condition I guarantees to find all output faults (as it covers all transitions of the FSM); (2) Condition II requires arc disjoint sequences for all transitions (if feasible) and both condition I and II require to visit all states after transition traversals; thus conditions I and II together are expected to discover most of the transfer faults (the actual fault coverage is investigated later in Section IV).

Building blocks of test sequences: In a nutshell, our algorithm uses a preamble part responsible for traversing all transitions of the FSM first, and then a postamble part responsible for traversing all states of the FSM to fulfill both conditions, but on different graphs. For condition I the original graph G (that corresponds to FSM M) will be used, for condition II different subgraphs of G can be selected when some t transitions are filtered out.

- *All Transition (AT):* This part specifies that all transitions of the given model should be covered at least once. This can be realized using the TT method without returning to the initial state at the end. Thus once all transitions are covered, the traversal of the resulting Euler tour stops.

- *All State (AS):* This part specifies that all states of a given model should be covered at least once. To find the shortest such sequence one can use a solution to the Traveling Salesperson problem [26], without the need to return to the initial state. Since the TSP problem is an NP hard problem, the Nearest Neighbour (NN) heuristic [12] is selected, which searches in each step for the closest unvisited state until such state exists.

ATS algorithm (high level view): To fulfill condition I, the AT and AS parts are generated in step 1 on graph G, respectively, then concatenated. The resulting sequence is called *main sequence* and it covers all transitions of the FSM and then visits all of its states.

For condition II, the AT and AS parts are created on different filtered subgraphs of G and then concatenated to generate appropriate *alternative sequences*. Then, these alternative sequences are applied one after the other. The standard version of our algorithm (denoted by ATS0) generates 2 alternative sequences in step 2 that are as arc disjoint as possible. Note that as these 2 alternative sequences do not necessarily meet condition II, an optional, iterative part is also presented in step 2.3 to provide additional alternative sequences. This iterative part terminates, if for all t transitions an arc disjoint sequence has been found (where it is feasible; ATSa version) or if a predefined iteration limit is reached (ATSx version). The output of the algorithm is a test suite that is the concatenation of the generated main sequence and the alternative sequences. The different versions of the ATS algorithm are summarized in Table I.

Table I
THE SUMMARY OF DIFFERENT ATS ALGORITHM VERSIONS

ID	notes	input	used graphs	output: test suite
ATS0	standard version	FSM M	original: G filtered: G^T, G^{ACT}	1 main sequence + 2 alternative seqs.
ATSa	version without iteration limit	FSM M	original: G filtered: G^T, G^{ACT}, G^{AC_k}	1 main sequence + max. $2n$ alternative seqs.
ATSx	version with iteration limit	FSM M, $depth$	original: G filtered: G^T, G^{ACT}, G^{AC_k}	1 main sequence + max. $depth + 2$ alternative seqs.

The 3 different versions (ATS0, ATSa, ATSx) of our algorithm allow the test engineer to find an appropriate trade-off between coverage and the length of the entire test suite. Note that after the detailed description a small scale example is presented to show step-by-step, how the algorithm works.

ATS algorithm (standard version, ATS0) :

STEP 1. Use AT and AS to create preamble and postamble subsequences, respectively on graph G. The concatenated preamble.postamble main sequence will guarantee that the test suite covers at least one walk from each transition to every state.

STEP 2. Create alternative sequences by concatenating the AT preamble and AS postamble subsequences generated on different subgraphs of G. To maintain the continuity of the entire sequence, each of the alternative sequences should start from the last state reached by the previous one.

STEP 2.1. For the first alternative sequence take the TS inverse spanning tree used in the Eulerian graph G_E during the execution of the AT part of step 1. Then, extend it with randomly selected shortest paths in G from the root node to each of the leaves of TS using breadth-first-search. This results in a strongly connected subgraph of G called G^T. The Eulerian augmentation of G^T is denoted with G_E^T used in AT preamble sequence generation. Then the postamble part is generated using AS on G^T.

STEP 2.2. For a second alternative sequence apply a filter on G that masks out the transitions belonging to G^T. This will be the complement graph of G^T, called G^{CT}. If G^{CT} is not strongly connected, then some transitions have to be reused from G^T, resulting in a graph G^{ACT}. The number of re-enabled transitions should be minimal in order to maintain the highest level of disjointedness using the following method:

STEP 2.2.1. $G^{ACT} := G^{CT}$. Let c denote the number of SCCs of G^{ACT}. Create a directed graph G_{SCC} with c number of nodes, each representing a distinct SCC of G^{CT}. Also create a $c \times c$ zero matrix A that denotes that each of the nodes of G_{SCC} are isolated at this stage.

STEP 2.2.2. For all i components of G_{SCC} check each outgoing arc of G from the nodes of component i and if it leads to component j (where $j \neq i$) and $A_{i,j} = 0$, then add an arc to G_{SCC} from the node representing component i to the node representing component j. Also set $A_{i,j} := 1$.

STEP 2.2.3. While $c > 1$:

STEP 2.2.4.1. Re-enable a random transition in the filter of G that connects two separate, previously unconnected components of G_{SCC} and add the corresponding arc to the filtered graph G^{ACT}.

STEP 2.2.4.2. Check for cycles in G_{SCC}, using depth-first search, if there is one, then merge the nodes that belong to the cycle into a single node representing a new larger SCC. Similarly, shrink the size of the corresponding A matrix. If h nodes were merged, then $c := c - (h - 1)$.

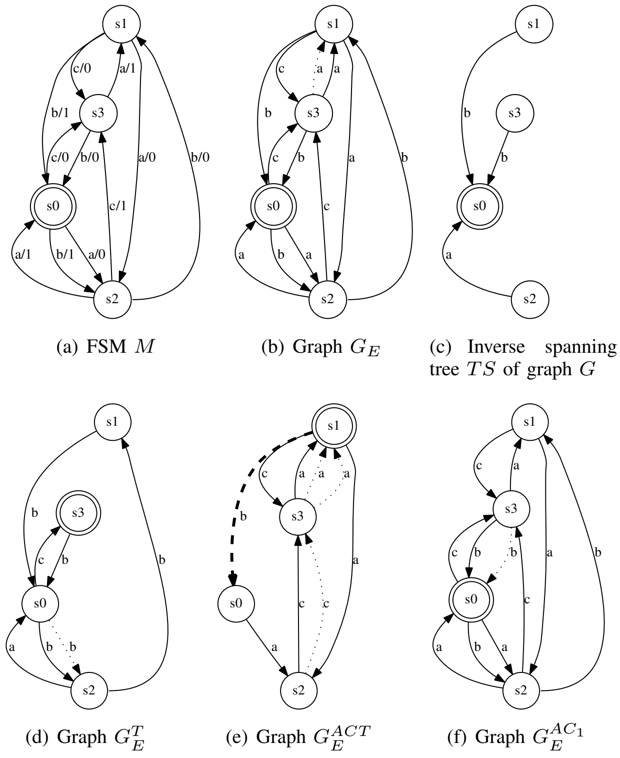
STEP 2.2.4. Once G^{ACT} is strongly connected again, generate preamble and postamble sequences using AT and AS, respectively.

Optional, iterative extensions for ATS (ATSa, ATSx): If transitions had to be re-enabled in step 2.2 to make G^{ACT} strongly connected, the alternative sequences generated for criterion II won't be entirely arc disjoint, i.e. criterion II is not met. In this case the following recursive part of graph filtering can be enabled:

STEP 2.3. The arcs that were both re-enabled in step 2.2

and in all previous iterations of step 2.3 (if there were previous iterations) are collected in the list arc_rem. Then, the arc_rem arcs are filtered out from G resulting in a graph G^{C_k} in the k^{th} iteration. Some of these arcs need to be re-enabled again to connect SCCs (similarly as in step 2.2) resulting in a subgraph G^{AC_k}. These re-enabled arcs remain in arc_rem list, the others are removed. Create an alternative sequence (by concatenating the appropriate AT preamble and AS postamble subsequences) on graph G^{C_k}. Run the function described above recursively until...

- no transitions remain in the list arc_rem or if the number of elements in arc_rem has not decreased since the previous step (ATSa).
- an iteration limit $depth$ is reached or the stop condition of ATSa is met (ATSx).



(a) FSM M (b) Graph G_E (c) Inverse spanning tree TS of graph G

(d) Graph G_E^T (e) Graph G_E^{ACT} (f) Graph $G_E^{AC_1}$

Figure 2. ATS example

ATS example: Here we demonstrate how our ATS algorithm works through a small scale example. We use the following notations in the figures: solid lines represent original arcs (i.e. the transitions of the FSM). Extra arcs, which make the graph balanced, are shown with dotted lines. The re-enabled transitions of filtered graphs that connect SCCs are shown with bold dashed lines. The initial state of each test sequence is denoted with a double circle. The input of each transition is also labeled on its corresponding arc in the graphs.

Consider FSM M in Figure 2(a). From this, an Eulerian graph G_E is created in step 1 – see Figure 2(b). The TS inverse spanning tree of G_E used by the AT part, when creating an Euler tour over G_E is shown in 2(c). The resulting AT input sequence of step 1 is *bbacacabaacb* starting at initial state s_0, followed by the AS input sequence *bbc* finishing at state s_3, forming a main sequence together. The first alternative sequence is created in step 2.1 using the G_E^T Eulerian graph of filtered graph G^T – see Figure 2(d). The resulting AT input sequence is *bbbbbac* starting at state s_3, followed by the AS input sequence *bbb* terminating at state s_1. The second alternative sequence is created in step 2.2 over G_E^{ACT} – see Figure 2(e). The resulting AT input sequence is *acabacac*[1], starting at state s_1, followed by the AS input sequence *aacab* terminating at state s_0. Here the standard version of the ATS algorithm (ATS0) terminates. Note that arc $s_1 \rightarrow s_0$ is re-enabled in G_E^{ACT} to connect two SCCs, i.e. it is used both in the first and the second alternative sequences. Thus, the iterative ATSa extension of the algorithm (described in step 2.3) can be enabled to create an arc disjoint sequence for the $arc\_rem = \{s_1 \rightarrow s_0\}$ element. At the first iteration, graph $G_E^{AC_1}$ is created – see Figure 2(f), and $s_1 \rightarrow s_0$ is removed from arc_rem. The corresponding AT part *bbacbaacac* starts at state s_0 and is followed by the AS part *aaa*. As $arc\_rem = \{\}$ the algorithm terminates.

ATS complexity calculation:

Standard version (ATS0): The complexity of the AT and AS generation parts are $O(n^3 + m)$ and $O(n^2)$, respectively, due to the TT and the NN algorithms. Thus, step 1 and step 2 require $O(n^3 + m)$ elementary steps, resulting in a total complexity of $O(n^3 + m)$ and in an $O(m)$ overall length for the test suite (in case of deterministic and completely specified FSMs $m = p \cdot n$, resulting in a $O(n(n^2 + p))$ complexity and $O(p \cdot n)$ length of the test suite).

Iterative extensions (ATSx and ATSa): The iterative part requires $O(\eta(n^3 + m))$ additional complexity, where $\eta < 2 \cdot n$ in case of the ATSa and $\eta \leq min(depth, 2 \cdot n)$ in case of the ATSx version, because subgraph G^T of step 2.1 contains no more than $2 \cdot (n-1)$ arcs (the TS inverse spanning tree contains exactly $n-1$ arcs, and the tree that contains the shortest path from the root node to each of the leaves of TS contains no more than $n-1$ arcs) that at worst case need to be filtered out. The total length of the resulting test suite is $O(\eta \cdot m)$.

As our ATS algorithm traverses all transitions of the FSM (AT part of step 1) it guarantees to find all output faults. As the algorithm traverses all transitions, then visits all states (step 1) and also provides alternative sequences that try to be as arc-disjoint as possible, then visit all states (step 2) it is expected to find most of the transfer faults; the actual fault coverage of different ATS algorithm versions (ATS0, ATSa, ATSx) are investigated in the next section.

[1] Note that the second extra multiplication of the $s_3 \rightarrow s_1$ arc is not used as all transitions are covered at least once when the algorithm visits state s_3 for the third time, so there is no need to finish the Euler tour with returning to start state s_1.

IV. Simulation results

We implemented our novel ATS algorithm, the random walk with 100% transition coverage stop condition, the TT and the HSI-methods in C++ using the graph algorithms and data structures of the LEMON[2] library.

The simulations were executed on a server running an Ubuntu 18.04.5 LTS operating system with 1 GB memory and one core of a shared Intel Xeon Gold 6140 CPU with 2.30GHz clock frequency.

We generated strongly connected, reduced random FSMs to investigate the performance of the algorithms. The strongly connected property is ensured by first creating a random inverse spanning tree, the arcs of which are directed towards the root node. Then a directed path is built from the root node that visits each of the leaf nodes. Finally, arcs are added between random nodes to reach the desired average outdegree denoted by $\overline{deg^+}$.

Table II
Investigated scenarios

		Number of states					
ID	CS / PS	min.	max.	size of step	$\overline{deg^+}$ / $\lvert I \rvert$	$\lvert O \rvert$	simulation goal
Scenario 1	PS	5	2000	5	5	5	complexity
Scenario 2	PS	5	800	5	25	5	complexity
Scenario 3	PS	5	100	5	5	2	fault cov.
Scenario 4	PS	5	100	5	5	5	fault cov.
Scenario 5	CS	5	2000	5	5	5	complexity
Scenario 6	CS	5	800	5	25	5	complexity
Scenario 7	CS	5	100	5	5	2	fault cov.
Scenario 8	CS	5	100	5	5	5	fault cov.

Different scenarios were created both for partially specified (PS) and completely specified (CS) FSMs[3] to investigate the complexity (time required for test generation and the size of the test suite) and the fault coverage of the algorithms – see Table II. In the last subsection the ATS algorithm is investigated on a small-scale telecommunication example.

A. Partially specified machines

1) Complexity investigations: Scenarios 1 and 2 examine how the time required for test generation and the overall length of the test sequences are affected by the number of states.

First, consider Scenario 1, where each state of the FSM has 5 transitions in average. Figure 3 shows the test generation time of the Random, TT and the ATS algorithms; the latter one with the standard version (ATS0), with the iterative versions with $depth$ parameters 1 (ATS1) and 2 (ATS2) and without a predefined $depth$ parameter (ATSa). The results indicate that the complexity of the TT and the ATS test generation is around the cubic function of the number of states. The test generation time of the Random algorithm is much less as it only selects a new transition randomly and checks if the stop condition is fulfilled at each step. Figure 4 shows the overall length of the resulting test sequences. As $\overline{deg^+}$ is fixed, the length of the test sequence is the linear function of the number of states. The length of the test sequence of ATS0, ATS1, ATS2 and ATSa is around 3.5, 4.7, 5.9 and 7 times longer on average as that of the one generated by the TT method, respectively.

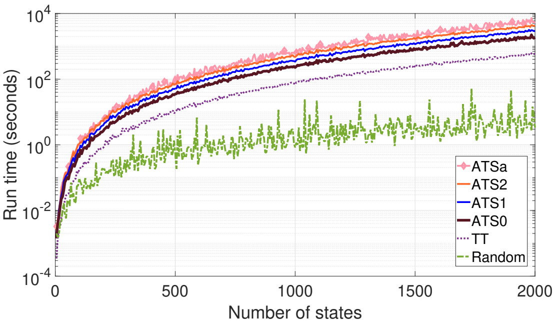


Figure 3. Scenario 1: Test generation time

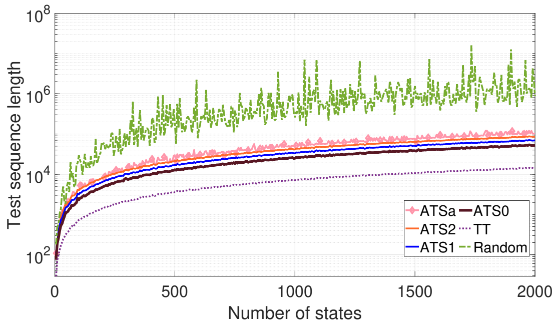


Figure 4. Scenario 1: Test sequence length

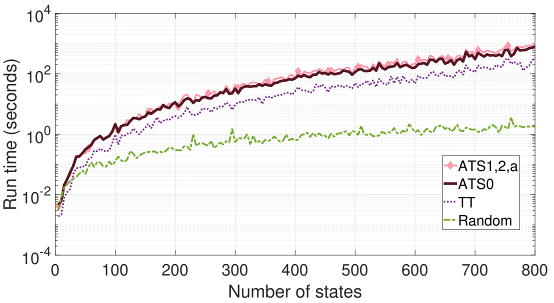


Figure 5. Scenario 2: Test generation time

We also investigate Scenario 2, when 25 transitions on average are set for each state of the FSM. Figures 5 and 6 show the test generation times and the overall lengths of the resulting test sequences, respectively. The trends are similar to the case of Scenario 1, but the complexities are higher due to denser FSMs. Also note that the ATS is able to create completely arc disjoint sequences in all cases even with 1 $depth$ parameter (ATS1) and if the number of states are relatively low, then even the standard version (ATS0) creates completely arc disjoint

[2] Library for Efficient Modeling and Optimization in Networks (LEMON), http://lemon.cs.elte.hu

[3] The motivation behind investigating both PS and CS machines with similar parameters is that the performance of the TT and ATS algorithms is expected to depend on how far each $s_j \in S$ state of the machine is from being balanced; in the latter case $deg^+(s_j) = \overline{deg^+} = |I|$ for all $s_j \in S$.

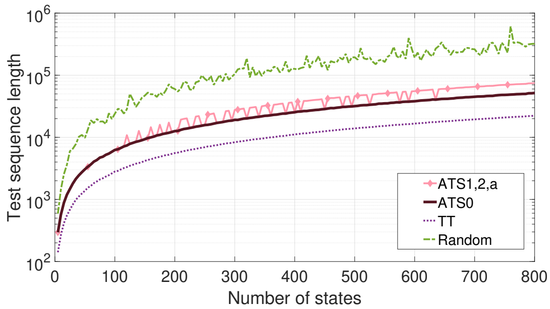


Figure 6. Scenario 2: Test sequence length

sequences. Due to this reason, the iterative extension of the ATS terminate earlier resulting in the same test generation times and lengths of test sequences for different ATS versions (ATS1, ATS2, ATSa). The length of the test sequence of ATS0 and ATS1/2/a is around 2.3 and 2.9 times longer on average as that of the one generated by the TT method, respectively.

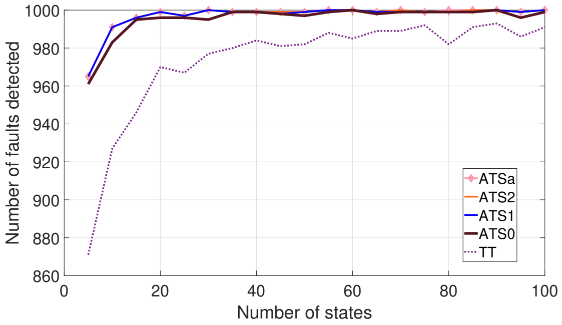


Figure 7. Scenario 3: Number of discovered faults

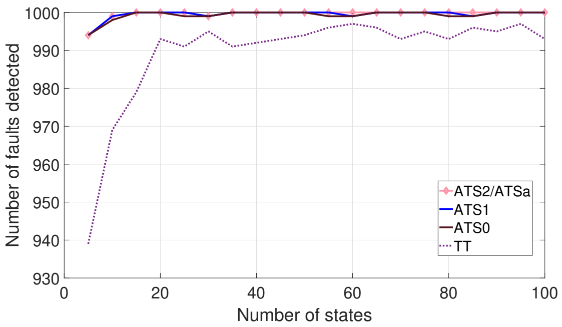


Figure 8. Scenario 4: Number of discovered faults

2) Fault coverage investigation: In Scenario 3 and 4 the fault coverage of different algorithms is investigated with randomly injected transfer faults[4] with 2 and 5 output symbols for the FSMs, respectively. Each data point in the figures had been obtained by 1000 simulation runs; in each simulation a single transition fault is injected to an FSM with given parameters and we observe how many times from these 1000 distinct cases do the algorithms discover the fault.

The results of the TT and the ATS method for Scenario 3 and 4 are presented in Figures 7 and 8, respectively. The results show that the ATS algorithm is much more effective in finding transfer faults than the TT, even with its standard version (ATS0). If the iterative part is switched on and the *depth* parameter increases or is switched off (ATS1 → ATS2 → ATSa), the fault coverage increases; for all but the smallest machines ATS1, ATS2 and ATSa is able to catch virtually all faults[5]. The relative number of discovered faults increases if the number of states increases both in case of the TT and of the ATS. The reason is that if the size of the test sequence increases, the probability that the desired output and the observed output of the test sequence differs increases. The difference between Scenario 3 and 4 simulations show that the probability of discovering faults increases as the number of output symbols is raised[6]. The reason is that different transitions with more possible output symbols to select from will more probably differ from each other.

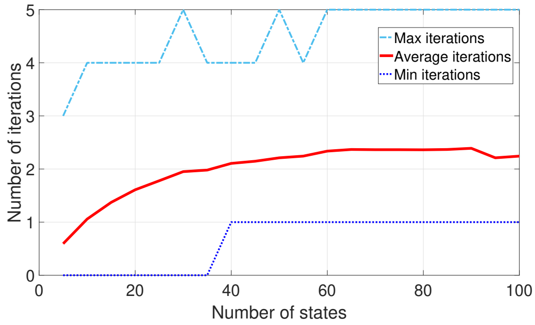


Figure 9. Scenario 3: Number of iterations of ATSa

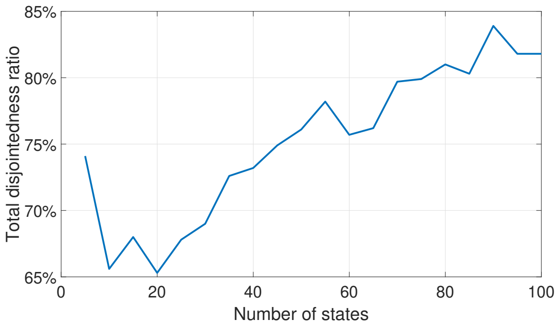


Figure 10. Scenario 3: Ratio of total arc disjoint test suites of ATSa

[4] Note that output faults are not investigated as the TT and the ATS algorithms traverse all transitions of the specification model, thus all of them are able to show both the absence or the presence of single output faults.

[5] Note that in Scenario 3 the fault coverage of ATS0 and ATS1 are almost identical in the performed simulations except at three points ($n = 45, 70, 85$) and that the fault coverage of ATS2 and ATSa only differs at one point ($n = 80$).

[6] Note that in Scenario 4 the fault coverage of ATS1 and ATS2 are almost identical in the performed simulations except at two points ($n = 60, 85$), while the fault coverage of ATS2 and ATSa is identical.

The minimum, the maximum and the average number of iterations for the ATSa algorithm version is also investigated – the results for Scenario 3 are presented in Figure 9[7]. For Scenario 3 Figure 10 presents the ratio when ATSa terminates because for all t transition an arc disjoint sequence has been found (in other cases for some transitions no arc disjoint sequence can be found due to the structure of the FSM)[7].

B. Completely specified machines

Similar scenarios were created for completely specified machines as in case of partially specified ones, but instead of average outdegree, we used the term number of input symbols, as for all states the number of outgoing transitions will be equal with this parameter.

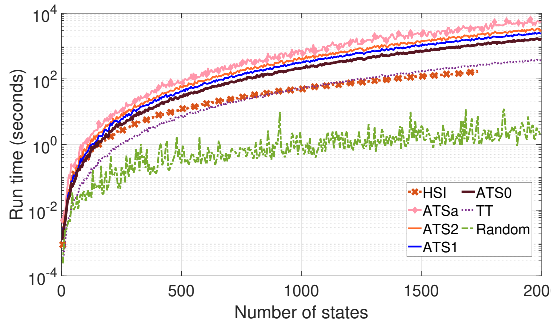


Figure 11. Scenario 5: Test generation time

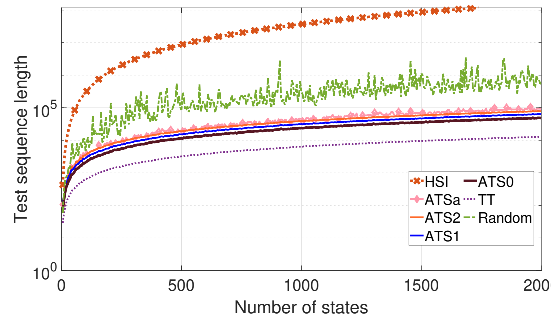


Figure 12. Scenario 5: Test sequence length

1) Complexity investigations: First consider Scenario 5, where the FSMs have 5 input symbols. Figure 11 and 12 show the test generation time and the entire length of the resulting test suite, respectively for the Random, HSI, TT algorithms and for the standard (ATS0) and iterative versions (ATS1, ATS2) of the ATS algorithm.

As in case of partially specified machines, the complexity of the TT and the ATS test generation is the cubic function of the number of states and the length of the TT and the ATS test sequences is the linear function of the number of states. Note that the test generation time of the TT and the different versions of ATS are about 35% and $15 - 22\%$ less than in case of their partially specified counterpart (Scenario 1), respectively. The reason is that in case of completely specified FSMs, every state has the same number of outgoing transitions, thus less extra arc multiplication is required in the Eulerian graph G_E of FSM M compared to the partially specified FSMs. For the same reason the overall length of the TT and the ATS test sequences are around 11% and 8-9% less in Scenario 5 compared to Scenario 1.

The test generation complexity is less than the theoretic cubic upper limit in case of the HSI method. The reason is that each member of the separating family of sequences typically consists of a test sequence with 1 or 2 length instead of the theoretical worst case $n - 1$ length. However, the size of the test suite generated by the HSI is significantly bigger than the ones generated by the TT and the ATS, as this test suite systematically checks all n states and $n \cdot (p - 1)$ remaining transitions of the FSM and the verification of a state or the end state of a transition requires $n - 1$ distinct sequences.

The test generation time and the entire length of the resulting test suite for FSMs with 25 input symbols are presented in Figure 13 and 14, respectively.

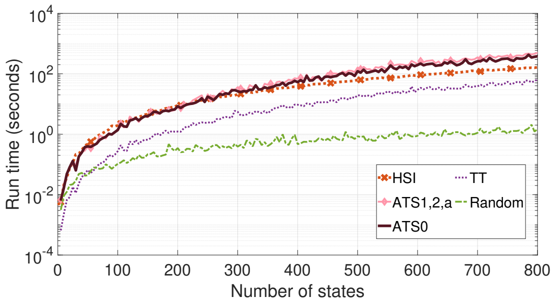


Figure 13. Scenario 6: Test generation time

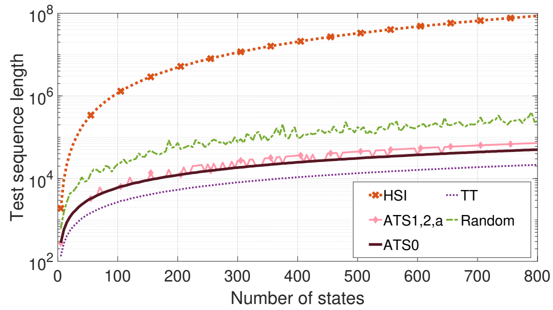


Figure 14. Scenario 6: Test sequence length

2) Fault coverage investigation: The results of the TT, the ATS and the HSI methods for Scenario 7 and 8 are presented in Figures 15 and 16, respectively. As expected, the structured HSI finds all next state faults and the TT-method discovers the least number of faults of the triple. The ATS algorithm is very efficient in discovering faults even with the standard version (ATS0) and it can be further enhanced if the iterative part is switched on (ATS1, ATS2 and ATSa). Note that in

[7] Note that the results are very similar for Scenario 4 as the output symbols of the transitions do not affect the test generation of ATS.

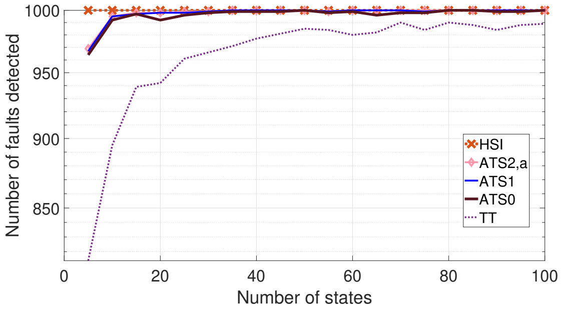


Figure 15. Scenario 7: Number of discovered faults

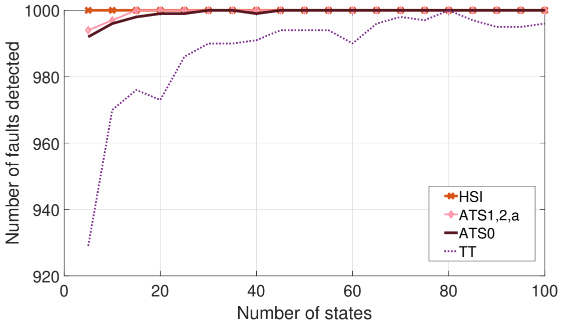


Figure 16. Scenario 8: Number of discovered faults

Scenario 8 ATS1, ATS2 and ATSa provide exactly the same fault coverage in the observed simulations and at and above 15 states they are able to discover all transfer faults as the HSI but with the fraction of its test suite size.

C. SIP UAC registration example

Simulations were also performed to investigate the ATS algorithm with an example from the telecommunication domain. For this, the following functionalities of the User Agent Client (UAC) during the registration process of the SIP (Session Initiation Protocol) [1] over the TCP (Transmission Control Protocol) transport layer were considered:

- Successful new registration (see Section 2.1 of [2])
- Cancellation of registration (see Section 10.2.2 of [1] and Section 2.4 of [2])
- Handle negative responses for registration requests (see Section 10.3 / $4^{th} - 6^{th}$ points of [1]).
- Interval too brief (see Section 10.3 / 7^{th} point of [1])
- Silent discard (see Section 10.2.7 / 7^{th} point of [1])
- Re-registration (see Sections 10.2.1.1 and 10.2.4 of [1])

The resulting FSM is presented in Figure 17. Note that only the signaling level was considered; a detailed description about how this FSM can be constructed from the related call-flows is presented in [21].

The length of the TT test sequence is 19 transitions, the overall length of the sequences generated by ATS0 is 47. Note that ATS0 algorithm can not find arc disjoint alternative sequences for three transitions and due to the structure of the FSM, the iterative version (ATSa) can not find arc disjoint sequences for these transitions, either.

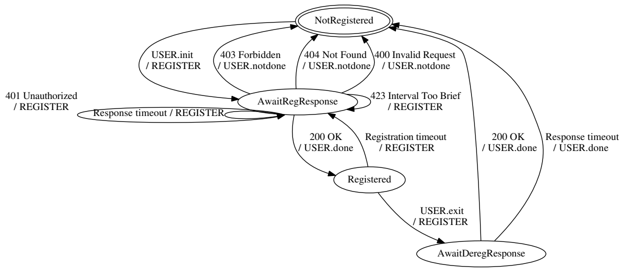


Figure 17. FSMs for the registration process of SIP UAC

As the FSM has 12 transitions and 4 states, $12 \cdot (4-1) = 36$ different atomic transition faults are possible; the corresponding 36 faulty FSMs were created and the fault coverage of TT and ATS was investigated. The TT was able to discover 32 and the ATS0 was able to find 35 faults.

V. Conclusion and Future Works

In the current article we proposed a new heuristic algorithm for the All-Transition-State criteria of deterministic finite state machine specifications. The length of the resulting test suite and its fault coverage can be fine-tuned with the three different versions of our algorithm (standard, iterative with and without an iteration limit) allowing the test engineer to find a suitable trade-off between the overall length of the test suite and fault coverage. The simulations show that the size of the resulting test suite has the same order of magnitude as the one produced by the TT-method, while its fault detection capability is near as effective as the one generated by the HSI-method, but with the fraction of its test suite size.

In the future we would like to extend the ATS algorithm to handle changing specifications, i.e. to identify the effects of changes in the test suite derived for a previous system version and to only update those parts that are necessary. As our algorithm reused some fundamental parts of the TT-method, many parts of the incremental TT [20] method can be utilized to fulfill this purpose. We also plan to extend our method for Extended Finite State Machine models, where the guarding conditions over variable values can also be considered when generating the test suite. We also would like to perform an extensive analysis with specification machines of different problem domains.

Acknowledgements

The project is supported by the Hungarian Government and co-financed by the European Social Fund (EFOP-3.6.3-VEKOP-16-2017-00001: Talent Management in Autonomous Vehicle Control Technologies).

The authors would like to thank László Dervalics for the implementation of the HSI-method.

REFERENCES

[1] RFC 3261: SIP: Session Initiation Protocol, 2002. https://tools.ietf.org/html/rfc3261 Accessed: 2021-09-02.

[2] RFC 3665: Session Initiation Protocol (SIP) Basic Call Flow Examples, 2003. https://tools.ietf.org/html/rfc3665 Accessed: 2021-09-02.

[3] Paul Ammann and Jeff Offutt. *Introduction to Software Testing*. Cambridge University Press, New York, NY, USA, 1st edition, 2008. DOI: 10.1017/CBO9780511809163.

[4] Gregor von Bochmann, Anindya Das, Rachida Dssouli, Martin Dubuc, Abderrazak Ghedamsi, and Gang Luo. Fault Models in Testing. In *Proceedings of the IFIP TC6/WG6.1 Fourth International Workshop on Protocol Test Systems IV*, pages 17–30, Amsterdam, The Netherlands, 1991. North-Holland Publishing Co.

[5] Eckard Bringmann and Andreas Krämer. Model-based testing of automotive systems. In *Proceedings of the 2008 International Conference on Software Testing, Verification, and Validation*, ICST '08, pages 485–493, Washington, DC, USA, 2008. IEEE Computer Society. DOI: 10.1109/ICST.2008.45.

[6] Manfred Broy, Bengt Jonsson, Joost-Pieter Katoen, Martin Leucker, and Alexander Pretschner (Eds.). *Model-Based Testing of Reactive Systems*. Springer, 2005. DOI: 10.1007/b137241.

[7] T. Chow. Testing software design modelled by finite-state machines. I*EEE Transactions on Software Engineering*, 4(3):178–187, May 1978. DOI: 10.1109/TSE.1978.231496.

[8] R. Geoff Dromey. Formalizing the Transition from Requirements to Design. In Zhiming Liu and Jifeng He, editors, *Mathematical Frameworks for Component Software: Models for Analysis and Synthesis*, pages 173– 205. World Scientific Series on Component-Based Development, 2006. DOI: 10.1142/9789812772831_0006.

[9] Jack Edmonds and Ellis L. Johnson. Matching, Euler tours and the Chinese postman. *Mathematical Programming*, 5(1):88–124, 1973. DOI: 10.1007/BF01580113.

[10] István Forgács and Attila Kovács. *Practical Test Design*. BCS, The Chartered Institute for IT, 2019.

[11] S. Fujiwara, G. v. Bochmann, F. Khendec, M. Amalou, and A. Ghedamsi. Test selection based on finite state model. *IEEE Transactions on Software Engineering*, 17(6):591–603, 1991. DOI: 10.1109/32.87284.

[12] Gregory Z. Gutin, Anders Yeo, and Alexey Zverovich. Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP. *Discret. Appl. Math.*, 117(1-3):81–86, 2002. DOI: 10.1016/S0166-218X(01)00195-0.

[13] Drago Hercog. Protocol Specification and Design. In *Communication Protocols*. Springer, Cham, 2020. DOI: 10.1007/978-3-030-50405-2_2.

[14] Gerard J. Holzmann. *Design and Validation of Protocols*. Prentice-Hall, 1990.

[15] David Lee and Mihalis Yannakakis. Principles and Methods of Testing Finite State Machines – A Survey. *Proceedings of the IEEE*, 84(8):1090–1123, 1996. DOI: 10.1109/5.533956.

[16] Y. Linand Y. C. Zhao. A new algorithm for the directed chinese postman problem. *Computers and Operations Research*, 15(6):577–584, 1988. DOI: 10.1016/0305-0548(88)90053-6.

[17] Gang Luo, Alexandre Petrenko, and Gregor V. Bochmann. Selecting Test Sequences for Partially-Specified Nondeterministic Finite State Machines. In *Proceedings of the IFIP WG6.1 7th International Workshop on Protocol Test systems* VI, pages 91–106. Springer, 1995. DOI: 10.1007/978-0-387-34883-4_6.

[18] Matheus Monteiro Mariano, Érica Ferreira de Souza, André Takeshi Endo, and Nandamudi Lankalapalli Vijaykumar. Comparing graph-based algorithms to generate test cases from finite state machines. *Journal of Electronic Testing*, 35(11–12):867–885, December 2019. DOI: 10.1007/s10836-019-05844-6.

[19] S. Naito and M. Tsunoyama. Fault detection for sequential machines by transition-tours. In *Proceedings of the 11th IEEE Fault-Tolerant Computing Conference (FTCS 1981)*, pages 238–243. IEEE Computer Society Press, 1981.

[20] Gábor Árpád Németh and Zoltán Pap. The incremental maintenance of transition tour. *Fundam. Inf.*, 129(3):279–300, July 2014. DOI: 10.3233/FI-2014-972.

[21] Gábor Árpád Németh and Péter Sótér. Teaching performance testing. *Teaching Mathematics and Computer Science*, 19(1):17–33, 2021. DOI: 10.5485/TMCS.2021.0518.

[22] S. C. Orloff. A Fundamental Problem in Vehicle Routing. *Networks*, 4:35–64, 1974. DOI: 10.1002/net.3230040105.

[23] Zoltán Pap, Mahadevan Subramaniam, Gábor Kovács, and Gábor Árpád Németh. A bounded incremental test generation algorithm for finite state machines. In *Proceedings of the 19th IFIP TC6/WG6.1 International Conference, and 7th International Conference on Testing of Software and Communicating Systems*, TestCom'07/FATES'07, pages 244–259, Berlin, Heidelberg, 2007. Springer-Verlag. DOI: 10.1007/978-3-540-73066-8_17.

[24] Volnei A. Pedroni. *Finite State Machines in Hardware. Theory and Design (with VHDL and SystemVerilog)*. The MIT Press, London, England, 2013. DOI: 10.7551/mitpress/9657.001.0001.

[25] Alexandre Petrenko, Nina Yevtushenko, Alexandre Lebedev, and Anindya Das. Nondeterministic state machines in protocol conformance testing. In *Proceedings of the IFIP TC6/WG6.1 Sixth International Workshop on Protocol Test Systems VI*, pages 363—378, NLD, 1993. North-Holland Publishing Co.

[26] J. B. Robinson. *On the Hamiltonian game (a traveling-salesman problem)*. RAND Corporation, Santa Monica, CA, 1949.

[27] M. Soucha and K. Bogdanov. Spyh-method: An improvement in testing of finite-state machines. *2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 194–203, July 2018. DOI: 10.1109/ICSTW.2018.00050.

[28] Mihalis Yannakakis and David Lee. Testing finite state machines: Fault detection. *Journal of Computer and System Sciences*, 50(2):209–227, 1995. DOI: 10.1006/jcss.1995.1019.

Gábor Árpád Németh obtained his MSc in Electrical Engineering and his PhD in Computer Science at the Budapest University of Technology and Economics (BME), Department of Telecommunication and Media Informatics (TMIT) in 2007 and 2015, respectively. He worked at Ericsson between 2011 and 2018 on a performance testing tool used in the telecommunication industry. Currently, he works at the Eötvös Loránd University (ELTE) on topics related to software testing.

Máté István Lugosi obtained his BSc in Computer Science at Eötvös Loránd University (ELTE) in 2021. Currently, he works at Ericsson on embedded software of microwave network devices. He studies in the MSc program of Computer Science at ELTE in the Cryptography specialization.

Security and Autonomic Management in System of Systems

Silia Maksuti, Mario Zsilak, Markus Tauber and Jerker Delsing

***Abstract*—A system of systems integrates systems that function independently but are networked together for a period of time to achieve a higher goal. These systems evolve over time and have emergent properties. Therefore, even with security controls in place, it is difficult to maintain a required level of security for the system of systems as a whole because uncertainties may arise at runtime. Uncertainties can occur from internal factors, such as malfunctions of a system, or from external factors, such as malicious attacks. Self-adaptation is an approach that allows a system to adapt in the face of such uncertainties without human intervention. This work outlines the progress made towards security mitigation in system of systems using a generic autonomic management system to assist engineers in developing self-adaptive systems. The manuscript describes the proposed system design, its implementation as part of the Eclipse Arrowhead framework, and its functionality in a smart agriculture use case. The system is designed and implemented in such a way that it can be reused and extended for a variety of use cases without requiring major changes.**

***Index Terms*—System of Systems, Security, Self-Adaptation, Autonomic Management, Eclipse Arrowhead**

I. Introduction

System of Systems (SoS) are large-scale integrated systems that can operate independently but are networked together for a period of time to achieve a higher goal, e.g., performance, robustness, security, etc. [1]. One of the main characteristics of SoS is the operational independence of the integrated systems. A system with low security level can compromise a system requiring high security level, and the compromise of such systems can lead to the compromise of the whole SoS, so security is an important concern.

Another characteristic of SoS is their distributed nature. In this manuscript, we use a drone-based application as an example of such a SoS. In [2], we proposed a use case for smart agriculture to assist winemakers and minimize travel time to remote and poorly connected infrastructures. The drone acts as a gateway by collecting sensor data and multispectral images of the vines and sending this data to a base station for offline analysis. In some cases, the drone is not always connected to the sensors and base station because the infrastructures are remote and poorly connected. Thus, it is a sporadically connected SoS where frequent changes may occur. If the security of one system (e.g., a wireless sensor network) is compromised, it may also affect the operation of other systems (e.g., the drone). Attackers can exploit these vulnerabilities to remotely control and disrupt the flow of data to/from the sensors and the drone. The ability to conduct a malicious attack on such systems can have serious consequences, and a large-scale, coordinated attack can disrupt national economies [3].

To establish a chain of trust between use case components, the Eclipse Arrowhead framework is used [4]. The goal of the framework is to efficiently support the development, deployment, and operation of SoS based on the fundamentals of service-oriented architecture (SoA): loose coupling, late binding, and lookup. The sensor nodes, drone, and base station are integrated into Arrowhead's local cloud through an automated onboarding procedure to ensure mutual authentication and thus secure communication [5]. A local cloud implements a set of services potentially used by all SoS applications.

While ensuring secure communication, the SoS should remain operational over a long period of time. To meet these requirements, the sensor nodes, the drone, and the base station must be optimally configured. However, due to the evolutionary development of SoS and emergent behavioral characteristics, ensuring these requirements can become a complex task. Uncertainties can occur due to internal factors (e.g., malfunction of a sensor node) or external factors (e.g., malicious attacks, weather conditions, etc.) that can affect secure communication between use case components. Even with mutual authentication, attackers can gain physical access to a sensor node and replicate many clones that have the same identity as the compromised node. The malicious node can then send additional sensor data to the drone. Similar behavior can occur when the sensor node malfunctions, such as when the battery is low. In this case, the sensor node cannot send enough data to the drone. To solve this problem, SoS must have mechanisms that allow them to self-adapt in the face of such uncertainties without human intervention.

In this manuscript, we propose to extend the smart agriculture use case with self-adaptation capabilities. We build on our previous work on the Generic Autonomic Management Framework [6, 7, 8] and extend it to support SoA-based frameworks as well. We propose a Generic Autonomic Management System (GAMS) to assist engineers in developing self-adaptive systems. Due to its generic nature, the system can be reused and extended for a variety of use cases without requiring major changes. This reduces the software engineering effort since the generic control mechanisms do not need to be (re)implemented for different use cases. A first concept of such a system is presented in [9]. In this manuscript, we present the design and implementation of a proof-of-concept for the proposed system and demonstrate its functionality in a smart agriculture use case.

The reminder of this manuscript is structured as follows. Section II reviews existing work on security and self-

DOI: 10.36244/ICJ.2021.3.7

adaptation in SoS. Section III provides the technical description of the smart agriculture SoS and motivates the need to extend the use case to include self-adaptation capabilities. Section IV describes the design and implementation of GAMS as part of the Eclipse Arrowhead framework. Section V presents the configuration of GAMS for the smart agriculture use case and experimental results. Section VI provides an overview of the results and future work.

II. Related Work

SoSs have several characteristics that distinguish them from traditional systems, such as the operational and managerial independence of their integrated systems, evolutionary development, emergent behavior, and geographic distribution. When designing a SoS, it is of utmost importance to understand the security implications of its features. For example, to address security-related aspects of a SoS that evolves over time and exhibits emergent characteristics, security mitigation approaches should be integrated. One approach is to augment the SoS with self-adaptive capabilities, as proposed in this manuscript. Therefore, we examine existing work on this topic.

Existing frameworks such as SASSY [10], MOSES [11], etc. have been developed to enable self-adaptation in service-oriented systems. Compared to these frameworks, our proposed system is intended to be generic so that it can be used in different SoA frameworks by a wide range of application systems without requiring a large amount of adjustments.

Ruz et.al. [12] have proposed a generic, self-adaptive framework to support monitoring and management tasks of component-based SoA applications. They separate the MAPE phases (Monitor, Analyze, Plan, and Execute) and implement them as distinct components that interact and support multiple sets of monitoring sources, conditions, policies, and distributed actions. Their main focus is on high scalability. Other works [13, 14, 15] justify the need for a generic solution for building self-adaptive systems. However, none of these works address security-related challenges that can be addressed by their solution.

Vishwa et al [16] have studied the adaptability of wireless sensor networks with the aim of highlighting the need for protection against malicious activities in such networks. They provide an evaluation of immune-based intrusion detection systems to determine that the functional requirements of wireless sensor networks such as self-organization, adaptability, fault tolerance, and self-healing are similar to human immune system mechanisms. The authors discuss the applicability of these theories to wireless sensor networks, and the paper ends with recommendations for expanding the study in the future.

There are other works dealing with SoS security such as [17], [18], [19], etc., which mainly focus on the engineering process that allows systems to integrate security at the design stage. They provide security artifacts (threats, attacks, assumptions about security properties, etc.) about the interaction with other elements or distributed systems to enable easy integration. In comparison, our work considers the evolving nature of SoS and its uncertainties that may arise at runtime, and proposes to address this problem by extending SoS with self-adaptation capabilities.

III. Smart Agriculture SoS

In this section, we present a use case from the smart agriculture domain, where the SoS approach is used to support winemakers and minimize travel time to remote and poorly connected infrastructures. In the following sections, we use this use case as a running example to show the functionality of the proposed system. An illustration of the use case is shown in Figure 1.

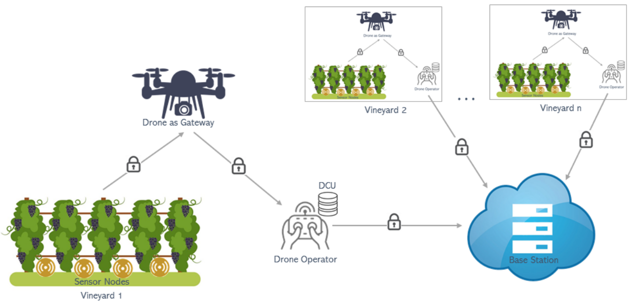


Fig. 1: High level view of the smart agriculture SoS use case.

A. Technical Description

The proposed SoS consists of a collection of sensor nodes connected via a wireless sensor network. These sensor nodes are strategically placed over the vineyard to obtain accurate measurements that can help detect diseases and conditions at early stages. The results of the sensor positioning are documented in [2]. The sensor nodes are equipped with sensors that collect environmental data such as air temperature, humidity and pressure, precipitation, wind speed and direction, sunlight, soil temperature and moisture, leaf wetness, etc. The sensor data is relayed to a drone, which acts as a gateway. The sensor nodes are constantly searching for the drone. When the drone is in range, a protected communication channel is established between the sensor nodes and the drone gateway. The drone sends sensor data from all sensor nodes and multispectral images of the vines to a base station for further analysis. However, in some cases, because the vineyards are located in harsh environments with poorly connected infrastructure, the sensor nodes first send the data to a data collection unit (DCU) located at the drone operator. Thus, this is a sporadically connected SoS. After the connection with the sensor nodes is established, the drone establishes a connection with the DCU and starts data transmission. The Wireless Local Area Network (WLAN) IEEE 802.11 is used for the communication link.

We have used the SysML modelling language to create a SysML block definition diagram for the smart agriculture use case, as shown in Figure 2.

1) Sensor Node: The sensor node consists of a single board computer, a Raspberry Pi, connected to several sensors, e.g. an air temperature and humidity sensor, a leaf wetness sensor, etc. The node has several *Python3* scripts that read the sensor data and write it to a comma-separated values (csv) file. Another *Python3* script continuously pings the drone gateway and transfers the csv files to the drone gateway if the connection is successful. The data transfer from the sensor nodes to the drone gateway is done using Hypertext Transfer Protocol Secure (HTTPs). For HTTPs, the communication protocol is encrypted using Transport Layer Security (TLS). After a successful file transfer, the file is moved to the archives and an entry is created in the log file.

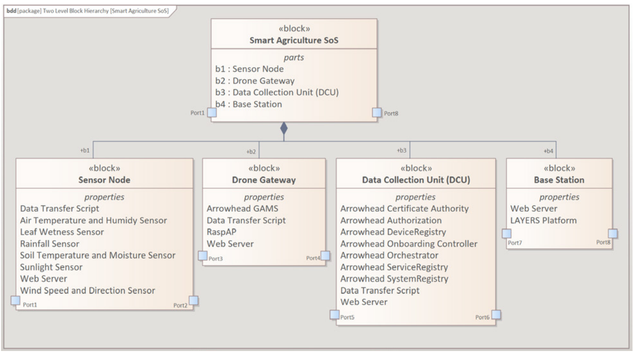


Fig. 2: SysML block definition diagram showing the hierarchy of the blocks composing the smart agriculture SoS use case.

2) Drone Gateway: The gateway mounted on the drone is a Raspberry Pi. RaspAP[1], a feature-rich wireless router software that works on Debian-based devices, is installed on the Raspberry Pi to configure the host access point daemon (hostapd). A web server receives the files sent by the sensor nodes. It also determines how the received files are handled, such as where they are stored. The gateway also contains a data transfer script similar to the script on the sensor node for post-processing the csv files. To ensure reusability and keep runtime resource costs low, both the web client on the sensor node and the web server on the drone gateway are written in *python*. To enhance the application with self-adaptation capabilities, the proposed Arrowhead support system (GAMS) is installed on the drone gateway. Details of its operation are described in Section IV.

3) Data Collection Unit: The DCU is a Debian system installed in a virtual machine. The supporting core systems of the Eclipse Arrowhead framework, which are invoked during the automated and secure onboarding procedure described later in the manuscript, are all installed on the DCU. In addition, a web server is installed to receive the files sent from the drone gateway and a data transfer script that sends them to the base station for further offline analysis.

4) Base Station: After the flight, the sensor data and the multispectral images stored in the base station are used for offline analysis. The LAYERS[2] platform is used to build a model and provide the necessary information about the condition of the vineyard. LAYERS is a platform that combines agronomic knowledge, earth observation remote sensing (drones, satellites, etc.) and artificial intelligence to create a proactive field monitoring system. It consists of a web tool with a map viewer and dashboard for field analysis, and an iOS and Android application for field sampling. However, the offline data analysis that takes place at the base station is beyond the scope of this manuscript.

B. Security and Adaptation Problem

Smart agriculture improves conventional farming methods by using sensors in vineyards to collect environmental data and autonomous vehicles (e.g., drones) to collect multispectral images of vineyards. These data are used for further offline analysis to improve production by optimizing crop management, such as accurate planting, irrigation, pesticide use, harvesting, etc. Despite the benefits, the use of internet-connected SoS can expose the agricultural sector to potential cyberattacks and vulnerabilities. Even if the vineyard is not connected to the Internet, it is an insecure network because an attacker only needs to be close enough to connect. These attacks can be used to remotely control and exploit sensors, actuators, and drones to destroy an entire field of standing crops, flood the vineyards, use smart drones to spray pesticides, etc. [20].

Therefore, it is of utmost importance to ensure trustworthy and secure communication of the drone with the sensor nodes in the vineyards and the base station. This ensures that only valid data is retrieved, damaged sensors are detected, and only authenticated and authorized systems participate in the communication. Even though all HTTP connections are secured via TLS (HTTPs), clients must be authenticated to ensure that sensor data is trusted.

To meet this security requirement, we use the automated and secure onboarding procedure of the Eclipse Arrowhead framework. An Arrowhead-compliant SoS is defined as a set of systems managed by the mandatory Arrowhead core systems that exchange information via services. Thus, a local cloud becomes an SoS. Also, two systems located in different local clouds and exchanging services form a SoS. The onboarding procedure [5] enables secure and trusted communication between such systems by using a chain of X.509 certificates generated at runtime. When a new device (e.g., a sensor node, drone, or base station) wants to interact with the Arrowhead local cloud, it should first authenticate itself using a valid preloaded Arrowhead certificate, manufacturer certificate, or shared key through the Onboarding Controller system. Each system hosted in this device will get a runtime certificate issued by Arrowhead. In Arrowhead, each local cloud has its own Certificate Authority (CA) system that issues and signs the runtime certificates of the systems. The CA system is the root of trust within the local cloud and can be signed by a central Arrowhead consortium, creating a chain of trust that allows different Arrowhead local clouds to be interconnected. Securely onboarding sensor nodes, drones, and base stations within the Arrowhead local cloud enables mutual authentication, allowing not only a TLS client to authenticate a server, but also a server to authenticate its client via X.509 certificates. The Arrowhead systems that are invoked during the onboarding procedure (ServiceRegistry, SystemRegistry, DeviceRegistry, Onboarding Controller, Orchestartor, Authorization, and Certificate Authority) are all located in the DCU. The source code and description of these systems can be found in the EclipseArrowhead GitHub[3] repository.

[1] https://raspap.com/ [2] https://hemav.com/en/services/digital-agriculture-en/

[3] https://github.com/eclipse-arrowhead/core-java-spring

This still leaves one attack vector open: clone attacks. Even with mutual authentication, attackers can gain physical access to a sensor node and replicate many clones with the same identity of the compromised node. The clones contain all the data of the legitimate sensor node and can successfully pass the onboarding procedure. Once the clones are on the network, they can exploit network operations such as routing, data collection, and key distribution, and even launch other attacks. This problem can be solved either by integrating secure elements into sensor nodes and drones (e.g., hardware security modules) to store keys and certificates in protected storage [21] or by extending the use case with self-adaptation capabilities that allow the system to adapt itself to a changing environment. The latter is described in the following sections.

IV. Generic Autonomic Management System

The Generic Autonomic Management System (GAMS) is designed and implemented as an Arrowhead support core system. A system is Arrowhead-compliant if it produces at least one service and consumes at least the three mandatory core services of the Eclipse Arrowhead framework, namely *ServiceDiscovery*, *AuthorizationControl* and *Orchestration* [4]. The *ServiceDiscovery* service is used to register and unregister services and to locate services among the registered services in the ServiceRegistry system. The *AuthorizationControl* service provides two different interfaces for retrieving authorization rights: (i) intra-cloud authorization, which defines an authorization right between a consumer and a provider system in the same local cloud for a particular service and (ii) inter-cloud authorization, which defines an authorization right for an external local cloud to consume a specific service from the local cloud. The *Orchestration* service provides application systems with orchestration information: where to connect. The output of this service includes rules that tell the application system which service provider systems to connect to and how (as a service consumer). Such orchestration rules include information about the reachability of a service provider (e.g., network address and port), service instance details within the provider system (e.g., base URL (Uniform Resource Locator), interface design specification, and other metadata), authorization-related information (e.g., access token and signature), etc.

The *GenericAutonomicManagement* service produced by GAMS is designed and implemented as a REST web service that can be invoked by different SoA-based frameworks. REST stands for representational state transfer and is a set of architectural constraints. Thus, a REST API is used for the interaction with the *GenericAutonomicManagement* service. A REST API is an application programming interface that conforms to the constraints of REST architectural style and enables interaction with REST web services [22]. The REST API has the following methods: (i) GET to retrieve information about the REST API resource, (ii) POST to create a REST API resource, (iii) PUT to update a REST API resource, and (iv) DELETE to delete a REST API resource. Compared to other protocols e.g. SOAP (Simple Object Access Protocol), REST APIs are faster and more lightweight for IoT applications [23].

A. System Description

We have used Systems Modeling Language (SysML) to create an internal block definition diagram of GAMS, as shown in Figure 3. The system enables autonomic control loops using MAPE-K (Monitor, Analyze, Plan, Execute and SharedKnowledge) as a reference feedback loop for self-adaptive systems [24]. An example of such interaction is a set of sensors and actuators (managed system), where GAMS is the autonomic manager (management system).

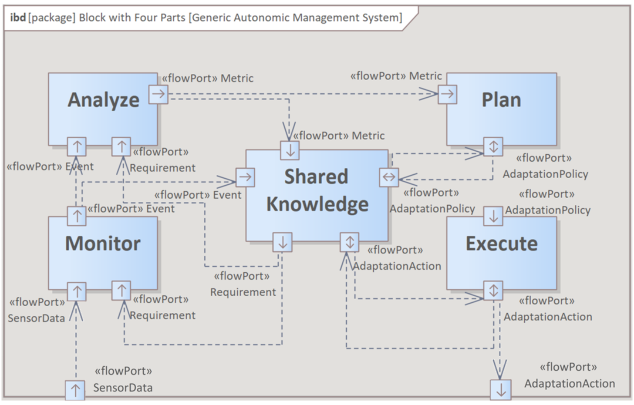


Fig. 3: SysML internal block definition diagram showing the internal structure of GAMS.

a) Monitor: The Monitor component continuously collects monitoring data from the sensor. The component performs a pre-analysis based on the incoming sensor data and the requirements stored in the SharedKnowledge. In case of a significant deviation, an event is generated and stored in the SharedKnowledge. The functions in this phase can aggregate the incoming data before passing it on to the next phase. GAMS allows you to specify the number of events to be considered for aggregation. If the specified number of events is not yet present, processing in this phase is aborted. The implemented functions are the following:

- **Sum:** creates a sum of sensor values.
- **Average:** creates an average of sensor values.
- **Trend:** indicates if the sensor values are increasing or decreasing.
- **Maximum:** uses the highest value in the next phase.
- **Minimum:** uses the lowest value in the next phase.
- **Count:** counts the number of incoming data in a specified time frame.
- **None:** does not aggregate the sensor value, but forwards it without change.

The SysML activity diagram of Monitor component is shown in Figure 4.

b) Analyze: The Analyze component evaluates the events received from the Monitor component with regard to the requirements and context data in the SharedKnowledge. If the requirements cannot be met, a change request is sent to the Plan component with a description of the metrics. Similar to the Monitor component, processing can be stopped if the result of the analysis shows that no action is required at this time. The implemented functions are the following:

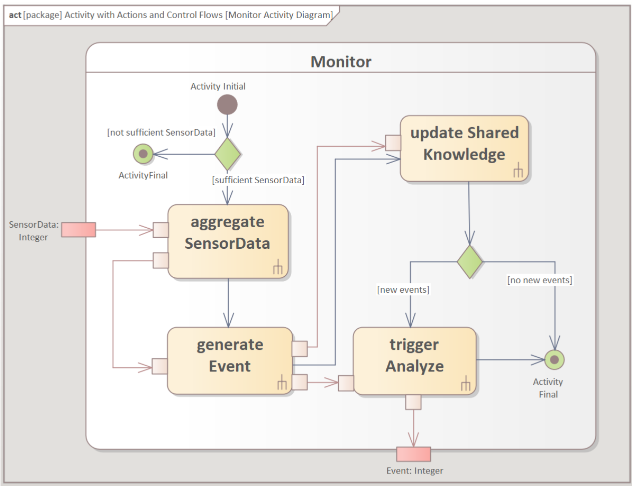


Fig. 4: SysML activity diagram showing the sequence of actions that are called as invocations of activities in the Monitor component of GAMS.

- **Count:** counts the number of generated events in a specified time frame.
- **Set-Point:** compares the incoming data with a configured target value (set-point). The target value could be a range with a lower and an upper set-point. In case both are used, this function acts as a double set-point.

The SysML activity diagram of Analyze component is shown in Figure 5.

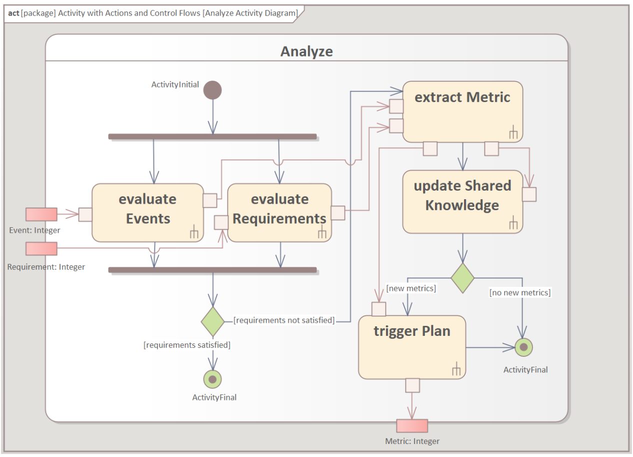


Fig. 5: SysML activity diagram showing the sequence of actions that are called as invocations of activities in the Analyze component of GAMS.

c) Plan: The Plan component is able to understand the metrics received from the Analyze component and to derive adaptation policies. It sets a corrective action and mandatory parameters for the autonomic element. An example of this is the use case of room temperature. A thermometer would be used as an input sensor and a heating or air conditioning system as an actuator. The planning component converts the incoming signal into a value that the actuator understands using the accumulated knowledge from the previous phases. The implemented functions are the following:

- **Match:** matches the incoming value as a key in a key-value structure and forwards the value to the next phase.
- **API Call:** makes an API Call to determine the value for the next phase.
- **Transform:** transforms the incoming value using a mathematical function.
- **None:** forwards the incoming value to the next phase without changing it.

The SysML activity diagram of Plan component is shown in Figure 6.

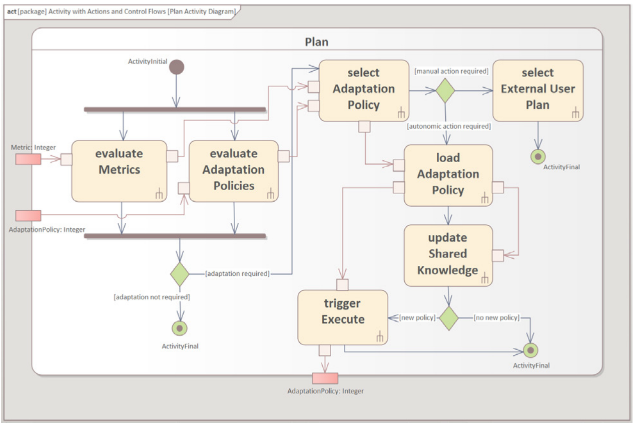


Fig. 6: SysML activity diagram showing the sequence of actions that are called as invocations of activities in the Plan component of GAMS.

d) Execute: The Execute component receives the policies from the Plan component and executes the derived action through the *GenericAutonomicManagement* service. The implemented functions are the following:

- **Composite Action:** allows the execution of multiple actions either in parallel or one after another.
- **API Call:** executes an API call as corrective action for the autonomic element.
- **Generate Event:** creates a new event to feed into the MAPE-K loop. This allows the re-evaluation of the sensor values with updated information from previous loops.
- **Logging Action:** logs the outcome of the MAPE-K loop.

The SysML activity diagram of Execute component is shown in Figure 7.

B. Service Interface Design Description

This section describes the HTTP/TLS/JSON *GenericAutonomicManagement* service interface. When a client request is made through a REST API, a representation of the state of the resource is transmitted to the GAMS endpoint. This information is transmitted in JSON format over HTTPs. Compared to other formats, JSON is language agnostic and can be read by both humans and machines [25].

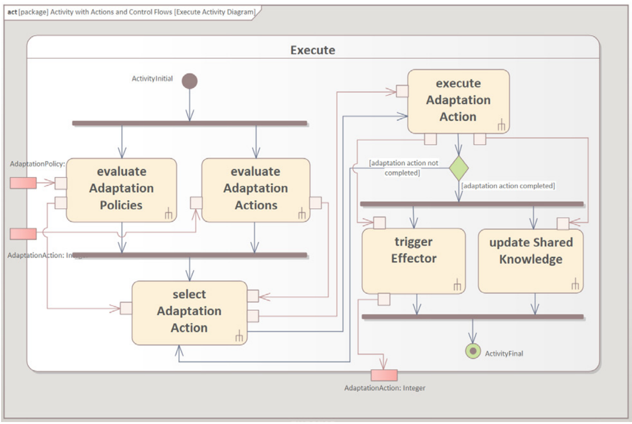
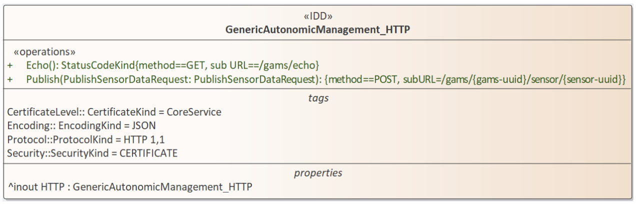

act [package] Activity with Actions and Control Flows [Execute Activity Diagram]
Execute
ActivityInitial
execute Adaptation Action
AdaptationPolicy:
evaluate Adaptation Policies
evaluate Adaptation Actions
[adaptation action not completed]
[adaptation action completed]
AdaptationAction: Integer
trigger Effector
update Shared Knowledge
select Adaptation Action
ActivityFinal
AdaptationAction: Integer

Fig. 7: SysML activity diagram showing the sequence of actions that are called as invocations of activities in the Execute component of GAMS.

1) Service Operations: In the following, an overview of the interface of the *GenericAutonomicManagement* service, its operations, data models and implementation is given and shown in Figure8. Both operations are described by their name and by an input type (between parenthesis) and an output type (at the end, preceded by a colon). Input and output types are only specified if they are accepted or returned by the interface in question.



«IDD»
GenericAutonomicManagement_HTTP

«operations»
+ Echo(): StatusCodeKind{method==GET, sub URL==/gams/echo}
+ Publish(PublishSensorDataRequest: PublishSensorDataRequest): {method==POST, subURL=/gams/{gams-uuid}/sensor/{sensor-uuid}}

tags
CertificateLevel:: CertificateKind = CoreService
Encoding:: EncodingKind = JSON
Protocol::ProtocolKind = HTTP 1,1
Security::SecurityKind = CERTIFICATE

properties
^inout HTTP : GenericAutonomicManagement_HTTP

Fig. 8: SysML block description diagram of the *GenericAutonomicManagement* service interface and its operations.

Publish(PublishSensorDataRequest) The Publish operation is used to send new sensor readings to the service, as exemplified in Listing 1. The sensor readings could be either numeric (integer or floating point number) or textual (event based), depending on the configuration. The sensor inputs feed the MAPE-K control loop of GAMS and eventually trigger a change on an actuator. The specific REST operation associated with this is: `POST/gams/{gams-uuid}/sensor/{sensor-uuid}`

```

POST /gams/f8c3de3d-1fea-4d7c-a8b0-29f63c4c3454/sensor/123
    e4567-e89b-42d3-a456-556642440000 HTTP/1.1

{
  "timestamp": "2021-07-04 12:00:00",
  "data": 1.2
}
```

Listing 1: An example of the Publish invocation for a floating point number.

Echo():StatusCodeKind The Echo operation returns an "is alive" response from the *GenericAutonomicManagement* service, as exemplified in Listing 2, which can be used to test the availability of the core service. The specific REST operation associated with this is: `GET/gams/echo`

```

GET /gams/echo HTTP/1.1

Got it!
```

Listing 2: An Echo invocation response.

Both operations respond with the HTTP status code `201 Created` when successfully invoked. The error codes are: `400 Bad Request` if request is incorrect, `401 Unauthorized` if an improper client certificate was provided, and `500 Internal Server Error` if *GenericAutonomicManagement* service cannot process the request due to an internal problem.

2) Information Model: The Publish operation has as input type the **PublishSensorDataRequest** structure, which is used to publish new sensor readings. The identification of the sensor is possible in two ways:

- Uniform Resource Identifier (URI) path parameters denoting the GAMS instance and the sensor identification,
- URI path parameter denoting the GAMS instance and using the source IP address to determine the sensor.

The POST request contains the parameters (data types) described in Table I.

Field	Type	Description
timestamp	DateTime	The date and time of the sensor reading. Pinpoints a specific moment in time.
data	SensorData	The value of the sensor reading. May be configured as integer number, floating-point number, or text string.

TABLE I: POST request parameters of the Publish operation.

The activity diagram shown in Figure 9 describes the process of publishing sensor data in GAMS.

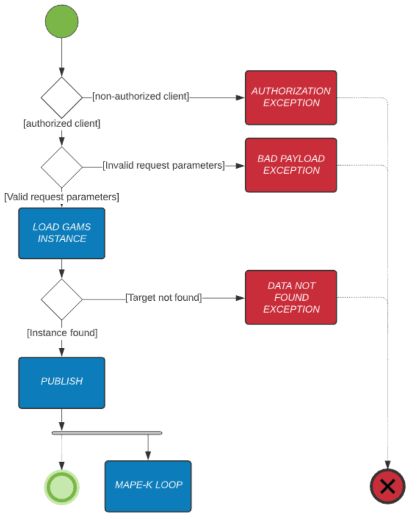


Fig. 9: Information model as an activity diagram that describes the process of publishing sensor data in GAMS.

To interact with the Arrowhead local cloud, each client must be authenticated and authorized through the onboarding procedure and register its device, systems and services respectively in the DeviceRegistry, SystemRegistry, and ServiceRegistry systems. When the client is properly authenticated and authorized, it receives the GAMS endpoint from the Orchestrator system. The client sends a POST request to the provided GAMS endpoint. If the POST request parameters are valid, the GAMS instance is loaded and the Publish operation is invoked.

V. GAMS INTEGRATED IN THE SMART AGRICULTURE SOS

In the smart agriculture use case described above, despite mutual authentication through the Arrowhead automated onboarding procedure, attackers can gain physical access to a sensor node and replicate many clones that have the same identity as the compromised node. The malicious node can spoof the media access control (MAC) address of the legitimate node to bypass possible security measures at the drone access point (e.g., MAC address whitelist). In addition, the malicious node can spoof the IP address of the legitimate node to bypass the IP address restriction on the client certificate, allowing the malicious node to send additional sensor data to the drone. Another problem can be caused by a real malfunction of a sensor node, such as a low battery. We propose to integrate GAMS into the smart agriculture use case so that the SoS can adapt itself in the face of such uncertainties.

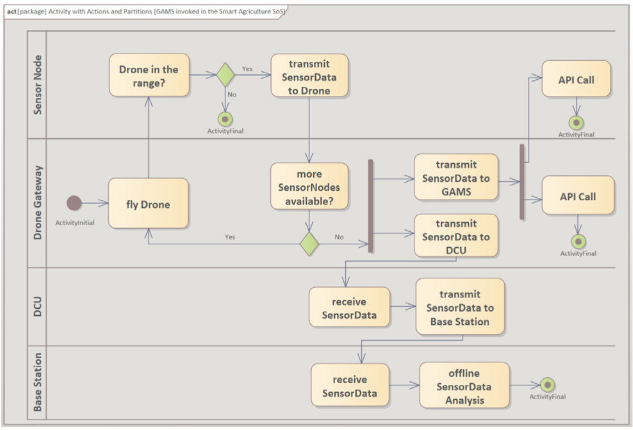

act [package] Activity with Actions and Partitions [GAMS invoked in the Smart Agriculture SoS]
Sensor Node
Drone Gateway
DCU
Base Station
Drone in the range?
Yes
No
ActivityFinal
transmit SensorData to Drone
API Call
ActivityFinal
ActivityInitial
fly Drone
more SensorNodes available?
Yes
No
transmit SensorData to GAMS
transmit SensorData to DCU
API Call
ActivityFinal
receive SensorData
transmit SensorData to Base Station
receive SensorData
offline SensorData Analysis
ActivityFinal

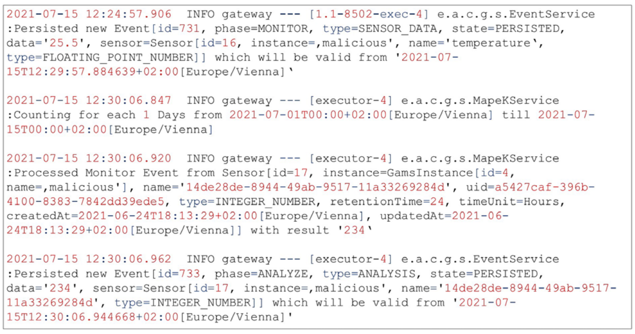
Fig. 10: SysML activity diagram that describes the process of invoking GAMS in the smart agriculture SoS use case.

The invocation of GAMS takes place in the drone gateway, as shown in Figure 10. The SharedKnowledge of GAMS contains information about traffic profiles, e.g., the expected number of sensor data, which serve as a baseline for normal traffic behavior. Such profiles may be provided by the industrial partners or generated from historical data. GAMS uses the sensor data as input for MAPE-K loop and compares the actual traffic to the baseline. It sends API calls when suspicious behaviors are detected that may indicate the following: (i) possible malicious attacks or (ii) sensor node malfunctions. An application-specific effector (or actuator) is then used to execute the adaptation action decided by GAMS (management system) in the drone and sensor nodes (managed system).

A. GAMS Configuration

GAMS is developed as a generic system; therefore, it should be reusable and extensible so that it can be applied to a variety of use cases without requiring major changes to the solution. As described in Section IV-A, multiple functions are implemented in each phase to cover a range of use cases. Configuring GAMS for a particular use case means that one or more functions from each phase will be selected and extended as needed, based on adaptation requirements.

a) Monitor: We have used the **Count** function implemented in the Monitor component. Each sensor node uses the Publish operation of *GenericAutonomicManagement* service interface to send sensor readings to GAMS. After receiving the sensor data, GAMS counts the number of sensor data for a configured time frame, in our use case it counts the number of sensor data per day.

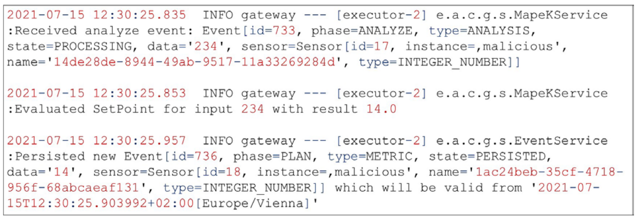


```
2021-07-15 12:24:57.906  INFO gateway --- [1.1-8502-exec-4] e.a.c.g.s.EventService
:Persisted new Event[id=731, phase=MONITOR, type=SENSOR_DATA, state=PERSISTED,
data='25.5', sensor=Sensor[id=16, instance=,malicious', name='temperature‘,
type=FLOATING_POINT_NUMBER]] which will be valid from '2021-07-
15T12:29:57.884639+02:00[Europe/Vienna]‘

2021-07-15 12:30:06.847  INFO gateway --- [executor-4] e.a.c.g.s.MapeKService
:Counting for each 1 Days from 2021-07-01T00:00+02:00[Europe/Vienna] till 2021-07-
15T00:00+02:00[Europe/Vienna]

2021-07-15 12:30:06.920  INFO gateway --- [executor-4] e.a.c.g.s.MapeKService
:Processed Monitor Event from Sensor[id=17, instance=GamsInstance[id=4,
name=,malicious'], name='14de28de-8944-49ab-9517-11a33269284d', uid=a5427caf-396b-
4100-8383-7842dd39ede5, type=INTEGER_NUMBER, retentionTime=24, timeUnit=Hours,
createdAt=2021-06-24T18:13:29+02:00[Europe/Vienna], updatedAt=2021-06-
24T18:13:29+02:00[Europe/Vienna]] with result '234‘

2021-07-15 12:30:06.962  INFO gateway --- [executor-4] e.a.c.g.s.EventService
:Persisted new Event[id=733, phase=ANALYZE, type=ANALYSIS, state=PERSISTED,
data='234', sensor=Sensor[id=17, instance=,malicious', name='14de28de-8944-49ab-9517-
11a33269284d', type=INTEGER_NUMBER]] which will be valid from '2021-07-
15T12:30:06.944668+02:00[Europe/Vienna]'
```

Fig. 11: The log file of the Monitor component.

The log file of the Monitor component is shown in Figure 11. In this example, it counts 234 SensorData/day and stores this as an event in the SharedKnowledge.

b) Analyze: We have used the **Set-Point** function implemented in the Analyze component. A Set-Point controller has two set points to which it switches the output. For our use case, the accepted range is between 180 and 220 SensorData/day. The Analyze component returns a positive number when the input exceeds the upper set point (POSITIVE_METRIC) and a negative number when the input is below the lower set point (NEGATIVE_METRIC). In all other cases, zero is returned (ZERO_METRIC). These metrics are forwarded to the Plan component.

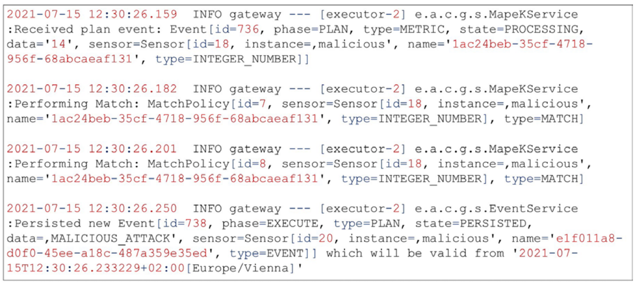


```
2021-07-15 12:30:25.835  INFO gateway --- [executor-2] e.a.c.g.s.MapeKService
:Received analyze event: Event[id=733, phase=ANALYZE, type=ANALYSIS,
state=PROCESSING, data='234', sensor=Sensor[id=17, instance=,malicious',
name='14de28de-8944-49ab-9517-11a33269284d', type=INTEGER_NUMBER]]

2021-07-15 12:30:25.853  INFO gateway --- [executor-2] e.a.c.g.s.MapeKService
:Evaluated SetPoint for input 234 with result 14.0

2021-07-15 12:30:25.957  INFO gateway --- [executor-2] e.a.c.g.s.EventService
:Persisted new Event[id=736, phase=PLAN, type=METRIC, state=PERSISTED,
data='14', sensor=Sensor[id=18, instance=,malicious', name='1ac24beb-35cf-4718-
956f-68abcaeaf131', type=INTEGER_NUMBER]] which will be valid from '2021-07-
15T12:30:25.903992+02:00[Europe/Vienna]'
```

Fig. 12: The log file of the Analyze component.

The log file of the Analyze component is shown in Figure 12. In this example it returns a POSITIVE_METRIC, since the input exceeds the upper set point with 14 SensorData/day.

c) Plan: We have used the **Match** function implemented in the Plan component. If an adaptation is required (POSITIVE_METRIC or NEGATIVE_METRIC), the Plan component selects a matching adaptation policy. If no adaptation is required (ZERO_METRIC), the operation is stopped.

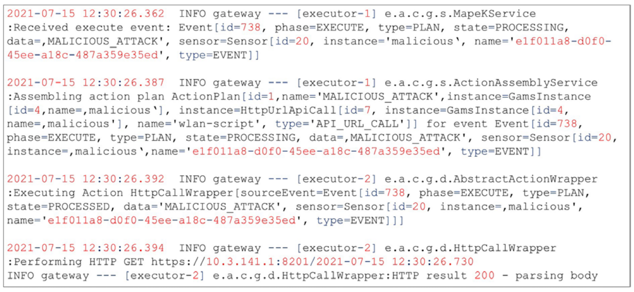


```
2021-07-15 12:30:26.159  INFO gateway --- [executor-2] e.a.c.g.s.MapeKService
:Received plan event: Event[id=736, phase=PLAN, type=METRIC, state=PROCESSING,
data='14', sensor=Sensor[id=18, instance=,malicious', name='1ac24beb-35cf-4718-
956f-68abcaeaf131', type=INTEGER_NUMBER]]

2021-07-15 12:30:26.182  INFO gateway --- [executor-2] e.a.c.g.s.MapeKService
:Performing Match: MatchPolicy[id=7, sensor=Sensor[id=18, instance=,malicious',
name='1ac24beb-35cf-4718-956f-68abcaeaf131', type=INTEGER_NUMBER], type=MATCH]

2021-07-15 12:30:26.201  INFO gateway --- [executor-2] e.a.c.g.s.MapeKService
:Performing Match: MatchPolicy[id=8, sensor=Sensor[id=18, instance=,malicious',
name='1ac24beb-35cf-4718-956f-68abcaeaf131', type=INTEGER_NUMBER], type=MATCH]

2021-07-15 12:30:26.250  INFO gateway --- [executor-2] e.a.c.g.s.EventService
:Persisted new Event[id=738, phase=EXECUTE, type=PLAN, state=PERSISTED,
data=,MALICIOUS_ATTACK', sensor=Sensor[id=20, instance=,malicious', name='e1f011a8-
d0f0-45ee-a18c-487a359e35ed', type=EVENT]] which will be valid from '2021-07-
15T12:30:26.233229+02:00[Europe/Vienna]'
```

Fig. 13: The log file of the Plan component.

The log file of the Plan component is shown in Figure 13. In this example, it returns MALICIOUS_ATTACK adaptation policy since it matches with the POSITIVE_METRIC.

d) Execute: We have used two functions implemented in the Execute component. The **API Call** function is used to trigger an effector, in our use case to invoke a service for changing the WLAN password in the drone and all sensor nodes, except the compromised node. The **Logging Action** function is used to create a log entry when the number of sensor data is below the lower set point.



```
2021-07-15 12:30:26.362  INFO gateway --- [executor-1] e.a.c.g.s.MapeKService
:Received execute event: Event[id=738, phase=EXECUTE, type=PLAN, state=PROCESSING,
data=,MALICIOUS_ATTACK', sensor=Sensor[id=20, instance='malicious‘, name='e1f011a8-d0f0-
45ee-a18c-487a359e35ed', type=EVENT]]

2021-07-15 12:30:26.387  INFO gateway --- [executor-1] e.a.c.g.s.ActionAssemblyService
:Assembling action plan ActionPlan[id=1,name='MALICIOUS_ATTACK',instance=GamsInstance
[id=4,name=,malicious‘], instance=HttpUrlApiCall[id=7, instance=GamsInstance[id=4,
name=,malicious'], name='wlan-script', type='API_URL_CALL']] for event Event[id=738,
phase=EXECUTE, type=PLAN, state=PROCESSING, data=,MALICIOUS_ATTACK', sensor=Sensor[id=20,
instance=,malicious‘,name='e1f011a8-d0f0-45ee-a18c-487a359e35ed', type=EVENT]]

2021-07-15 12:30:26.392  INFO gateway --- [executor-2] e.a.c.g.d.AbstractActionWrapper
:Executing Action HttpCallWrapper[sourceEvent=Event[id=738, phase=EXECUTE, type=PLAN,
state=PROCESSED, data='MALICIOUS_ATTACK', sensor=Sensor[id=20, instance=,malicious',
name='e1f011a8-d0f0-45ee-a18c-487a359e35ed', type=EVENT]]]

2021-07-15 12:30:26.394  INFO gateway --- [executor-2] e.a.c.g.d.HttpCallWrapper
:Performing HTTP GET https://10.3.141.1:8201/2021-07-15 12:30:26.730
INFO gateway --- [executor-2] e.a.c.g.d.HttpCallWrapper:HTTP result 200 - parsing body
```

Fig. 14: The log file of the Execute component.

The log file of the Execute component is shown in Figure 14. In this example, it executes a HttpCallWrapper adaptation action, which is the API call associated with MALICIOUS_ATTACK adaptation policy.

The log files presented in this section illustrate only one example of suspicious behavior that can be detected by GAMS, namely a possible malicious attack due to an increased number of SensorData/day. Another example is a possible malfunction of a sensor node due to a decreased number of SensorData/day. In this case, GAMS would return a NEGATIVE_METRIC corresponding to the MALFUNCTION_SENSOR adaptation policy and create a log entry as an adaptation action.

B. Results

The experimental environment consists of a *testDataset* generator script written in *python3*, a legitimate node containing valid sensor data readings (generated by the *testDataset* script), and a clone containing invalid sensor data readings (generated by the *testDataset* script). The invalid sensor data readings typically shows a higher temperature (+10°C) and contains more sensor data entries than the upper set point (220 SensorData/day) stored in the SharedKnowledge. The rest of the environment uses the actual scripts from the use case as described in Section III-A.

The measurements are simulated for 10 days. On the first and second day, the legitimate node sends valid data. From the third day, the clone with the same credentials as the legitimate node sends invalid data. The drone flies every other day. When GAMS is invoked, it receives the sensor data collected by the drone and detects an increased number of sensor data exceeding the upper limit defined in the SharedKnowledge on the third and fourth day. According to the configuration described in SectionV-A, GAMS sends an API call to the drone and the legitimate node to change the WLAN password. From this point on, the clone can no longer connect to the drone, so only valid data is sent. For testing purposes, we used only two nodes. However, in a real scenario, both the clone and the compromised node do not receive the new password to connect. The results are shown in Figure 15.

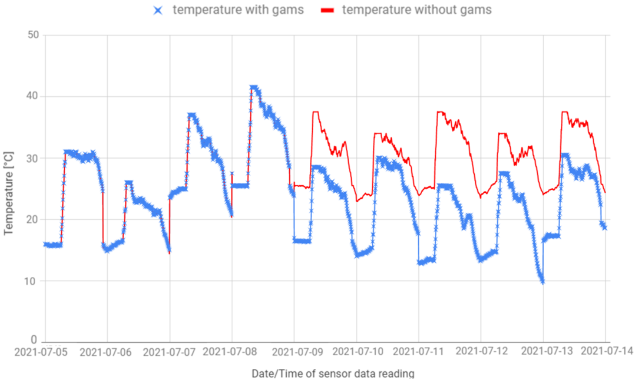


Fig. 15: The use case execution with/without GAMS invoked.

The results indicate that GAMS is able to detect unexpected changes in a SoS and take adaptation actions without human intervention, which can help maintain required security levels for the use case even in the presence of uncertainty.

VI. Conclusion

SoSs evolve over time and exhibit new properties, such that security controls introduced in the design phase to mitigate potential attack vectors may not be appropriate or sufficient later in operation. In this manuscript, we justify the need to provide SoS with self-adaptive capabilities to address security issues that may arise from uncertainties that are difficult to predict before the system is deployed. Such uncertainties may stem from factors internal or external to the SoS.

To address this challenge, we proposed a generic autonomic management system (GAMS) that automatically tracks runtime uncertainties and adapts SoS settings without human intervention. The internal building blocks of GAMS (Monitor, Analyze, Plan, Execute, and Shared-Knowledge) are designed and implemented in such a way that they can be reused and extended for a variety of use cases without requiring major changes. This reduces the software engineering effort. We integrated GAMS into a smart agriculture use case to demonstrate its functionality. The results showed that GAMS is able to detect a change in the environment and successfully send an API call to the drone and sensor nodes to change system settings to mitigate a potential malicious attack or detect a sensor node malfunction.

As future work, we plan to improve the codebase of GAMS to increase performance for resource-constrained systems, and evaluate it in various use cases to show its generic property.

ACKNOWLEDGMENT

This work has received funding from ECSEL Joint Undertaking (JU) under grant agreement No 826610 Comp4Drones project. The JU receives support from EU's Horizon 2020 research and innovation programme and Spain, Austria, Belgium, Czech Republic, France, Italy, Latvia, Netherlands.

REFERENCES

[1] Mark W Maier. Architecting principles for systems-of-systems. *Systems Engineering: The Journal of the International Council on Systems Engineering*, 1(4):267– 284, 1998. DOI: 10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D.

[2] Silia Maksuti, Michael Pickem, Mario Zsilak, Anna Stummer, Markus Tauber, Marcus Wieschhoff, Dominic Pirker, Christoph Schmittner, and Jerker Delsing. Establishing a chain of trust in a sporadically connected cyber-physical system. In *2021 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pages 890–895. IEEE, 2021.

[3] Sina Sontowski, Maanak Gupta, Sai Sree Laya Chukkapalli, Mahmoud Abdelsalam, Sudip Mittal, Anupam Joshi, and Ravi Sandhu. Cyber attacks on smart farming infrastructure. In *2020 IEEE 6th International Conference on Collaboration and Internet Computing (CIC)*, pages 135–143. IEEE, 2020. DOI: 10.1109/CIC50333.2020.00025.

[4] Jerker Delsing. *Iot automation: Arrowhead framework*. Crc Press, 2017.

[5] Ani Bicaku, Silia Maksuti, Csaba Hegedűs, Markus Tauber, Jerker Delsing, and Jens Eliasson. Interacting with the arrowhead local cloud: On-boarding procedure. In *2018 IEEE industrial cyber-physical systems (ICPS)*, pages 743–748. IEEE, 2018. DOI: 10.1109/ICPHYS.2018.8390800.

[6] Markus Tauber. Autonomic management in a distributed storage system. arXiv preprint arXiv:1007.0328, 2010.

[7] Silia Maksuti, Ani Bicaku, Markus Tauber, Silke Palkovits-Rauter, Sarah Haas, and Jerker Delsing. Towards flexible and secure end-to-end communication in industry 4.0. In *2017 IEEE 15th International Conference on Industrial Informatics (INDIN)*, pages 883–888. IEEE, 2017. DOI: 10.1109/INDIN.2017.8104888.

[8] Silia Maksuti, Oliver Schluga, Giuseppe Settanni, Markus Tauber, and Jerker Delsing. Self-adaptation applied to mqtt via a generic autonomic management framework. In *2019 IEEE International Conference on Industrial Technology (ICIT)*, pages 1179–1185. IEEE, 2019. DOI: 10.1109/ICIT.2019.8754937.

[9] Silia Maksuti, Markus Tauber, and Jerker Delsing. Generic autonomic management as a service in a soa-based framework for industry 4.0. In *IECON 2019 – 45th Annual Conference of the IEEE Industrial Electronics Society*, volume 1, pages 5480–5485. IEEE, 2019. DOI: 10.1109/IECON.2019.8927245.

[10] Daniel Menasce, Hassan Gomaa, Joao Sousa, et al. Sassy: A framework for self-architecting service-oriented systems. *IEEE software*, 28(6):78–85, 2011. DOI: 10.1109/MS.2011.22.

[11] Valeria Cardellini, Emiliano Casalicchio, Vincenzo Grassi, Stefano Iannucci, Francesco Lo Presti, and Raffaela Mirandola. Moses: A framework for qos driven runtime adaptation of service-oriented systems. *IEEE Transactions on Software Engineering*, 38(5):1138–1159, 2011. DOI: 10.1109/TSE.2011.68.

[12] Cristian Ruz, Françoise Baude, and Bastien Sauvan. Flexible adaptation loop for component-based soa applications. In *Seven International Conference on Autonomic and Autonomous Systems*, 2011.

[13] Sylvain Frey, Ada Diaconescu, David Menga, and Isabelle Demeure. Towards a generic architecture and methodology for multi-goal, highly-distributed and dynamic autonomic systems. In *10th International Conference on Autonomic Computing ({ICAC} 13)*, pages 201–212, 2013.

[14] Mahdi Ben Alaya and Thierry Monteil. Frameself: an ontology-based framework for the self-management of machine-to-machine systems. *Concurrency and Computation: Practice and Experience*, 27(6):1412–1426, 2015. DOI: 10.1002/cpe.3168.

[15] Svein Hallsteinsen, Kurt Geihs, Nearchos Paspallis, Frank Eliassen, Geir Horn, Jorge Lorenzo, Alessandro Mamelli, and George Angelos Papadopoulos. A development framework and methodology for self-adapting applications in ubiquitous computing environments. *Journal of Systems and Software*, 85(12):2840–2859, 2012.

[16] Vishwa T Alaparthy, Amar Amouri, and Salvatore D Morgera. A study on the adaptability of immune models for wireless sensor network security. *Procedia computer science*, 145:13–19, 2018. DOI: 10.1016/j.procs. 2018.11.003.

[17] Jose Fran Ruiz, Carsten Rudolph, Antonio Maña, and Marcos Arjona. A security engineering process for systems of systems using security patterns. In *2014 IEEE International Systems Conference Proceedings*, pages 8–11. IEEE, 2014. DOI: 10.1109/SysCon.2014.6819228.

[18] J Dahmann, George Rebovich, Michael McEvilley, and G Turner. Security engineering in a system of systems environment. In *2013 IEEE International Systems Conference (SysCon)*, pages 364–369. IEEE, 2013. DOI: 10.1109/SysCon.2013.6549907.

[19] Larry B Rainey and Andreas Tolk. Modeling and simulation support for system of systems engineering applications. 2015. DOI: 10.1002/9781118501757.

[20] Maanak Gupta, Mahmoud Abdelsalam, Sajad Khorsandroo, and Sudip Mittal. Security and privacy in smart farming: Challenges and opportunities. *IEEE Access*, 8:34564–34584, 2020. DOI: 10.1109/ACCESS.2020.2975142.

[21] Dominic Pirker, Thomas Fischer, Christian Lesjak, and Christian Steger. Global and secured uav authentication system based on hardware-security. In *2020 8th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud)*, pages 84–89. IEEE, 2020. DOI: 10.1109/MobileCloud48802.2020.00020.

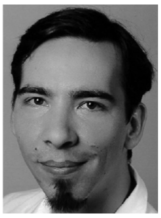
[22] Mark Masse. *REST API Design Rulebook: Designing Consistent RESTful Web Service Interfaces*. " O'Reilly Media, Inc.", 2011.

[23] Snehal Mumbaikar, Puja Padiya, et al. Web services based on soap and rest principles. I*nternational Journal of Scientific and Research Publications*, 3(5):1–4, 2013.

[24] Jeffrey O Kephart and David M Chess. The vision of autonomic computing. *Computer,* 36(1):41–50, 2003. DOI: 10.1109/MC.2003.1160055.

[25] Nurzhan Nurseitov, Michael Paulson, Randall Reynolds, and Clemente Izurieta. Comparison of json and xml data interchange formats: a case study. *Caine*, 9:157–162, 2009.

DI Silia Maksuti is a PhD student at Luleå University of Technology, Sweden, and works as a researcher at the University of Applied Sciences Burgenland, Austria, in the research center ”Cloud and Cyber Physical Systems Security”. Recently, she was working at the Austrian Institute of Technology (AIT) in the AIT’s ICT-Security Program. She received the Dipl-Ing. degree in Communication Engineering from the Carinthia University of Applied Sciences, Klagenfurt, Austria, and her B.Sc. degree in Telecommunication Engineering from the Polytechnic University of Tirana, Albania. She has been part of several EU projects, e.g., SECCRIT, SEMI40, PRODUCTIVE4.0, ArrowheadTools and Comp4Drones.



Mario Zsilak works as software engineer at the Center for Cloud and CPS Security at the Forschung Burgenland GmbH. He has contributed to Arrowhead in a number of projects. He is currently completing his Master in the MSc Program Business Process Management and Engineering at the University of Applied Sciences Burgenland. He completed his BSc in Information and Communication Systems and Services in 2017, at the University of Applied Sciences Technikum Wien.

Prof. (FH) Dr. Markus Tauber works as Chief Scientific Officer at Research Studios Austria Forschungsgesellschaft. Between 2015 until 2021, he worked as FH-Professor for the University of Applied Sciences Burgenland, where he held the position: director of the MSc program “Cloud Computing Engineering” and led the research center “Cloud and Cyber-Physical Systems Security”. From 2012 until 2015, he coordinated the “High Assurance Cloud” research topic at the Austrian Institute of Technology (AIT) part of AIT’s ICT-Security Program. Amongst other activities, he was the coordinator of the FP7 Project “Secure Cloud computing for CRitical infrastructure IT” - (www.seccrit.eu) and involved in the ARTEMIS Project Arrowhead. From 2004 to 2012, he was working at the University of St Andrews (UK), where he worked as a researcher on various topics in the area of network and distributed systems and was awarded a PhD in Computer Science for which he was working on “Autonomic Management in Distributed Storage Systems”.

Prof. Jerker Delsing received the M.Sc. in Engineering Physics at Lund Institute of Technology, Sweden 1982. In 1988 he received the PhD. degree in Electrical Measurement at the Lund University. During 1985 - 1988 he worked part time at Alfa-Laval - SattControl (now ABB) with development of sensors and measurement technology. In 1994 he was promoted to associate professor in Heat and Power Engineering at Lund University. Early 1995 he was appointed full professor in Industrial Electronics at Lulea University of Technology where he currently is the scientific head of EISLAB, http://www.ltu.se/eislab. His present research profile can be entitled IoT and SoS Automation, with applications to automation in large and complex industry and society systems. Prof. Delsing and his EISLAB group has been a partner of several large EU projects in the field, e.g. Socrades, IMC-AESOP, Arrowhead, FAR-EDGE, Productive4.0 and Arrowhead Tools. Delsing is a board member of ARTEMIS, ProcessIT.EU and ProcessIT Innovations.

AutoML for Log File Analysis (ALFA) in a Production Line System of Systems pointed towards Predictive Maintenance

Matthias Maurer, Andreas Festl, Bor Bricelj, Germar Schneider, and Michael Schmeja

Abstract—**Automated machine learning and predictive maintenance have both become prominent terms in recent years. Combining these two fields of research by conducting log analysis using automated machine learning techniques to fuel predictive maintenance algorithms holds multiple advantages, especially when applied in a production line setting. This approach can be used for multiple applications in the industry, e.g., in semiconductor, automotive, metal, and many other industrial applications to improve the maintenance and production costs and quality. In this paper, we investigate the possibility to create a predictive maintenance framework using only easily available log data based on a neural network framework for predictive maintenance tasks. We outline the advantages of the ALFA (AutoML for Log File Analysis) approach, which are high efficiency in combination with a low entry border for novices, among others. In a production line setting, one would also be able to cope with concept drift and even with data of a new quality in a gradual manner. In the presented production line context, we also show the superior performance of multiple neural networks over a comprehensive neural network in practice. The proposed software architecture allows not only for the automated adaption to concept drift and even data of new quality but also gives access to the current performance of the used neural networks.**

Index Terms—**Arrowhead Tools, AutoML, Log Analysis, Neuronal Architecture Search, Predictive Maintenance Framework**

I. INTRODUCTION

Predictive Maintenance (PdM), which roots can be traced back to 1940, gained more and more attention with the rise of automated data acquisition and data processing in decision making [30]. By monitoring system parameters, such as performance, vibrations, temperature development, oil conditions, noise generation, or the like, a useful purpose should be derived. The promises associated with using PdM are diverse, starting with management control, reduction of overtimes, reduction of downtimes, higher quality output, higher user support, etc. [27]. As diverse as the desired benefits of PdM are, the systems' type, under which predictive maintenance is applied, might even be more diverse. This type, besides general principles of PdM, needs to be considered when designing an appropriate PdM approach.

One of the regarded system types, under which PdM is applied, includes an ever-changing production line in a System of Systems (SoSs). An SoS, as a construct of systems, where each was designed and can be used for a main purpose other than being part of this SoS [2, 25], usually brings along non harmonized log messages and uncoordinated behaviour. When combining these different systems with an everchanging environment, e.g., due to a replacement of certain subsystems or because of a changing system load, a highly untransparent and difficult to predict SoS is created. Depending on the concrete setup of the system, different PdM approaches can be applied, among these are log analysis approaches.

Log analysis is a rather easy approach to apply to predictive maintenance [31] since in most cases log data, as a basis for the predictive maintenance intervention, is produced automatically and no further adjustments to the system are needed. They get collected anyway and, hence, only need processing to yield useful predictive maintenance findings. Log analysis approaches can be found for software [1, 12, 13, 15, 24, 28] or hardware SoSs [32], following different PdM objectives with different means. Among these, one can find visual tree representation [15], prediction heuristics, such as the so-called Dispersion Frame Technique [24], or machine learning methods [4, 13].

This work was submitted for review on the 14th of June 2021.

This research work has been funded by the European Commission, through the European H2020 research and innovation programme, ECSEL Joint Undertaking, and National Funding Authorities from 18 involved countries under the research project Arrowhead Tools with Grant Agreement no. 826452. The publication was written at Virtual Vehicle Research GmbH in Graz and partially funded within COMET Competence Centers for Excellent Technologies from the Austrian Federal Ministry for Climate Action, the Austrian Federal Ministry for Digital and Economic Affairs, the Province of Styria (Dept. 12) and the Styrian Business Promotion Agency (SFG). The Austrian Research Promotion Agency (FFG) has been authorised for the programme management.

Matthias Maurer is with the 'Contextual Information Systems & Operational Insights' group at the Virtual Vehicle Research GmbH, Graz, Austria (email: matthias.maurer@v2c2.at).

Andreas Festl is Senior Researcher for Data Science at Virtual Vehicle Research GmbH, Graz, Austria.

Bor Bricelj is a Senior Researcher and Data Scientist, working with the "Information Network Extraction Systems" group at the Virtual Vehicle Research GmbH, Graz, Austria. His work is focused on domains of computer vision and data enrichment.

Germar Schneider is Senior Expert for Factory Integration at Infineon Technologies Dresden GmbH & Co. KG and work package leader in the Arrowhead Tools project, Dresden Germany.

Michael Schmeja is the Area Manager for Safety and Security at the Virtual Vehicle Research GmbH, Graz, Austria.

DOI: 10.36244/ICJ.2021.3.8

Machine learning, as one approach to log analysis, is itself a well-researched academic area with application in nearly every imaginable area, especially in autonomous driving, health care, finance, manufacturing, and energy harvesting [3]. It is generally divided into supervised, unsupervised, and reinforcement learning [19]. Supervised learning uses features to predict labels, unsupervised learning uses features to get an insight about their statistical properties, and reinforcement learning uses a reward system for the current model behaviour to optimize the model's behaviour. Depending on the maintenance task in mind, an appropriate approach needs to be identified. In terms of identifying situations that need maintenance actions in advance, one can use the system's logs as features and the maintenance situations as labels, which are expressed by certain log entries. This would suggest supervised learning as a suitable method approach. Supervised learning methods include logistic regression, decision trees, support vector machines, and neural networks, among others [4].

A special fascination holds automated machine learning (AutoML) in this context since it could enable a PdM system to adapt to a changing environment. AutoML, which is mainly used in natural language processing (NLP) and computer vision (CV), aims at automating the entire pipeline of machine learning. Although there have been major achievements in NLP and CV, other areas are neglected [14]. This is also true for log analysis, which would benefit twice from such an approach. Firstly, such an automatization would provide access to this technology for a wider audience and, in general, support the creation of better ML systems. Secondly, besides these general advantages, this would allow the PdM system to update its outdated ML components automatically whenever it is necessary due to the changing environment.

The contributions of this paper are the following: a PdM framework for a steadily changing production line is introduced and different NN architectures are evaluated against each other within this framework using a proof-of-concept implementation. The practical relevance and automated nature of the approach allow for wide applicability, especially for novices in the area of machine learning.

We will now show in this paper the high potential of AutoML in the context of a production line system of systems. Therefore, we first discuss general AutoML techniques in Section 2, before we discuss the applications of AutoML in production systems of systems in Section 3. Section 4 shows the first implementation of the theoretical ideas of Section 3. Finally, Section 5 discusses the conclusions based on this work and possible further work in this context.

II. GENERAL AUTOML TECHNIQUES

An extensive review of the state-of-the-art regarding AutoML in the context of neuronal networks (NN) was done by Xin He, Kaiyong Zhao, and Xiaowen Chu [14]. They describe the AutoML pipeline consisting of four stages: data preparation, feature engineering, model generation, and model evaluation. Data preparation, as a means to obtain useful data, is composed of data collection, cleaning, and augmentation. Based on these steps, feature selection, extraction, and construction are used during feature engineering to obtain the features from the data, which are later used for model generation. This model generation can itself be divided into search space, where the ML model's structure is defined (e.g. support vector machine, k-nearest neighbours, neural networks) and optimization methods, which are concerned with optimizing hyperparameter and architecture of the previously defined model. Model evaluation, as the last described stage, is used to evaluate a model's performance. The AutoML pipeline is shown in Fig. 1.

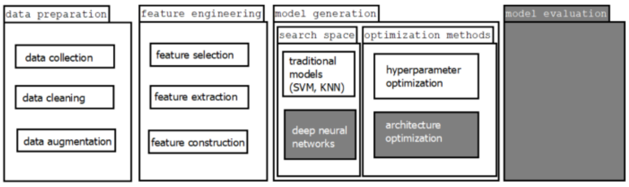


Fig. 1. AutoML pipeline as described by Xin He, Kaiyong Zhao, and Xiaowen Chu [14]. The elements shown in grey describe NAS elements.

Parts of this AutoML pipeline are referred to as neural architecture search (NAS), a sub-topic of AutoML gaining increased attention most recently [14]. NAS includes architecture optimization in the case of a neural network search space from the area of model generation in combination with model evaluation. The idea is to create a basic NN ML model based on the considered search space by applying architecture optimization to this structure and to create the final model by hyperparameter optimization. Evaluation during this procedure is inevitable. The search space describes how candidates for the model's basic structure are found and can be entire-structured, cell-based, hierarchical, and morphism-based. Entire-structured approaches create a structure by selecting layers and their order from a pool of layer candidates. Cell-based approaches use a fixed number of repeating cell structures, consisting of different blocks, which are concatenated afterward and consist itself of different layers combined at the end. One can tune the model by selecting the number of blocks, the operations of the layers in a block, and the combination method at the end of a block (e.g. addition, concatenation, etc.) and cell. An exemplary cell structure is shown in Fig. 2.

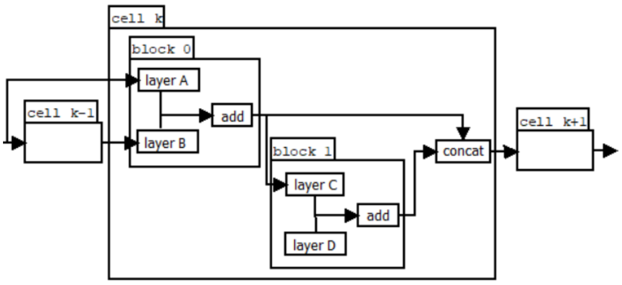


Fig. 2. Exemplary cell structure as shown by Xin He, Kaiyong Zhao, and Xiaowen Chu [14].

In comparison to cell-based approaches, hierarchical search also focuses on the network structure and not only on the cell structure. There are different approaches, all allowing for a fitting on a cell level and a network level [20, 21, 22]. Morphism-based search space uses already existing model structures to improve already existing networks and, hence, create new networks [33].

After defining the NN based on the search space, architecture optimization is used to find the best-performing architecture, which always includes the evaluation of different NNs. This search for the best architecture can be regarded as a search for a hyperparameter, where human expertise is needed. Different algorithms aim at automating this process, such as grid/random search, the evolutionary algorithm, reinforcement learning, gradient descent, surrogate model-based optimization, and hybrid methods. Grid and random search are two very basic optimization methods not considering any feedback from the current state of the architecture and might be considered a baseline approach for comparison. The evolutionary algorithm is a heuristic optimization algorithm, which uses an evaluation procedure until a stopping criterion is met. Starting with a set of NN, the evaluation procedure, inspired by biological evolution, selects a subset based on the NNs' performance, generates a new network from every two previously selected NNs, mutates the resulting NN a bit, and removes the worst-performing new NNs. Another recognized approach, reinforcement learning [37], usually uses a recurrent neural network (RNN) to incrementally improve the architecture by executing certain actions leading to a so-called reward influencing the next action and moving in that manner through the search space. This is shown in Fig. 3.

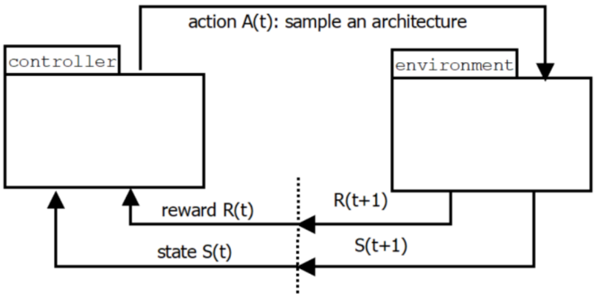


Fig. 3. Basic functionality of reinforcement learning as shown by Xin He, Kaiyong Zhao, and Xiaowen Chu [14].

In comparison to these already mentioned methods to search for the best-performing architecture, gradient descent is an approach allowing for a continuous search space [14, 23]. For that reason, it uses a continuous relaxation of the architecture representation, which is then used for optimization for the operations used in one node of the architecture's cell and leads to one architecture. Surrogate model-based optimization is a broadly used approach for architecture optimize by building a surrogate model to predict the best performing architecture [8, 17, 26]. Of course, all mentioned methods might be combined in a hybrid approach.

After deciding on the architecture to use, which is done with the same set of hyperparameters in most cases, one can turn to the optimization of the hyperparameters for the used architecture [14]. Therefore, different approaches are used, such as grid/random search, Bayesian optimization, and gradient-based optimization. Alternatively, hyperparameter and architecture optimization (HAO) can be used to optimize hyperparameter and architecture in combination [34].

All NAS approaches share one mutual problem, which is the need for frequent evaluation of architectures, leading to a high need for time and computing resources [14]. Different approaches, such as weight sharing [29], surrogate methods [9], early stopping [7], and low fidelity methods [18], aim at reducing this need for resources.

III. ADVANTAGES OF AUTOML IN A PRODUCTION SETTING

AutoML has proven its usefulness in the context of NLP [5, 16] and CV [11, 35], other areas have been neglected [14]. Especially in a production setting, applying AutoML approaches to log data might be useful, since they not only bring along the known advantages, such as easy access to NNs and improvement of ML models but also advantages specific to a production setting. Since there is a wide variation in production settings, we will now describe one rather generic production setting to show the usefulness of AutoML approaches when working with log data in this context.

One aspect of the production line setting is the usage of log data, which is produced by the subsystems and generally easily accessible - merely a central collection of this already existing and accessible data is required for the proposed utilization in a NN. Naturally, one can collect a unique ID for a specific log entry, a unique ID for a specific subsystem where the log entry originated, the time of occurrence, and possibly a duration. Depending on the actual setup further data might be recorded. In this setting, some log IDs might have an informing character, others might indicate a critical or interesting situation in the overall system. It is of utmost importance to prevent the cause of critical log entries from happening or, if this is not possible, to quickly react to the negative influences associated with such a critical log entry. In a predictive maintenance manner, a NN can be trained to predict upcoming log entries of interest-based on the observed log entries to allow for appropriate intervention.

Another aspect of the hereinafter regarded production line setting is its SoS nature, which determines its composition of distinct, changing subsystems working on a changing production load. Both addressed circumstances, a changing system and a changing production load are realistic due to continuous improvements of the production line and variations in the production demands, and they impact heavily on the log data. An altered production load influences the statistic properties of the log data and, hence, might render a trained NN unsuitable. These changing statistical properties, denoted as concept drift [10], are not necessarily happening incremental but might happen suddenly without any further indication on how the change might unfold, due to an unforeseen change in production. An even greater influence is exhibited by a new subsystem, which might introduce a new quality of log data, which cannot be handled by a trained NN.

The described problems of a sudden concept drift change and the introduction of a new quality of log data, specific to the described production line setting, can be addressed by AutoML approaches. Concept drift, as the first addressed problem, is already a discussed topic in literature [10]. Forgetting mechanisms, for example, allow to incorporate data with different emphasis, depending on how recent they are. In combination with change detection based on sequential analysis, statistical process control, distribution comparison, or contextual approaches, the software can react to concept drift and an adjustment of the model can be initiated. This adjustment, or learning, can be divided into two approaches: retraining, where the current model is discarded, and incremental adaption, where the current model is updated. Hence, concept drift is a phenomenon one can cope with, whereas an introduction of a new quality of log data, as the second addressed problem, is not yet discussed on a wide basis. When working with an embedding layer to map the log IDs to n-dimensional vectors, for example, one would have to adjust the vocable size (number of different log IDs) as input to this layer. This would require the creation and training of a new model.

One alternative approach for introducing new log IDs to a NN, without adapting the NN, is feature hashing [6]. This would lead to classes of IDs, which are used in the model. These classes would be indistinguishable for the model, new IDs would be assigned to one of the already existing classes. Over the course of time, when new ids occur and old IDs vanish, this might lead to a situation where log entries of one ID are used to predict the occurrence of an entirely other log ID. Also, two IDs, which are very different in their behavior, might be bundled together in one class. Such a bundling would, hence, be problematic in certain cases.

By introducing NNs for each log ID of interest, one would be able to create an overall ML model capable to adapt to concept drift concerning individual log IDs and it would be able to introduce a NN based on a new log ID as soon as enough data has been recorded to train such a NN. This allows for an incremental adaption of the overall model, even if new log IDs are introduced. However, each NN would need to accept unknown log IDs as feature values in a residual category. As soon as the respective NN gets retrained, this alarm ID does not receive a separate appearance in the NN. In the background, the active NNs need to be evaluated, replacement candidates are trained using AutoML methods and compared to the active NNs. Whenever a replacement candidate outperforms an active NN, the replacement candidate is incorporated into the overall model. The described workflow can be found in Fig. 4.

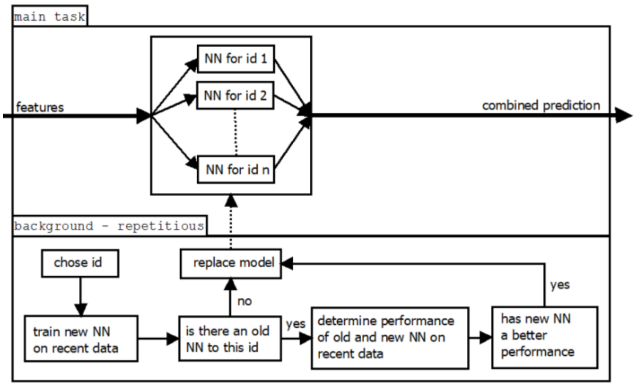

main task
features
NN for id 1
NN for id 2
NN for id n
combined prediction
background - repetitious
chose id
replace model
yes
no
train new NN on recent data
is there an old NN to this id
yes
determine performance of old and new NN on recent data
has new NN a better performance

Fig. 4. Overall concept of a prediction model based on multiple NNs, each concerned with predicting one ID.

Another advantage of using log data in the described setup is that labels can be calculated directly based on the features. Input features of the NNs are log IDs, device IDs, times, and so on, output labels are log IDs of interest. This means that the true labels are received sometime after the prediction, which allows for an evaluation of the current NNs and training of replacement candidates.

IV. Analysis of Proof-of-concept Implementation

Automated decision-making by predictive diagnosis and machine learning is a main topic within the Arrowhead Tools project, a Horizon 2020 project aiming for digitalization and automation solutions for the European industry. In this project different partners from the industry develop new tools to improve the European industry by creating many different new tools e.g., the Arrowhead Framework, but also tools based on new algorithms or neuronal networks which are used in different use cases. One important use case for complex maintenance tasks is the work on equipment data in the semiconductor industry. We worked together with the company Infineon Technologies Dresden on a use case using neuronal networks for a better understanding of the failure in a highly complex wafer transportation system to create predictive maintenance solutions saving time and high personal efforts. Another goal is, that this setup could be used in many different other industrial applications showing high potentials for interoperability. A first implementation of the theoretical ideas discussed in the previous section is implemented. We will first explain the available data, continue with the chosen NN structure, and finish with the used software design.

Available Data Quantities

The log entries in this AHT use case exhibit unique IDs, a start timestamp, an end timestamp, and a spatial location expressed as a segment of spatially close positions. This information is available for all entries, except for the segment, which is available only for around 50% of the data. Therefore, an additional category was introduced, expressing that no location information is available.

Based on these available log data quantities, different derived quantities can be constructed, such as observed log IDs, time since log occurrence, the active state of a log entry, and the segment of occurrence. For a given point in time, these quantities can be fed into a NN – the last log entries, each expressed as ID, time since the occurrence, the information, if it is still active, and the segment of occurrence. This can either be done for a fixed number of past log entries or, more accurately, for a fixed time in the past. To ensure a fixed-size input length to the NN in the latter case, such a fixed number of considered log entries need to be defined. If there are more log entries falling in the designated timeframe, they are ignored and if there are fewer log entries falling in this timeframe, placeholder values need to be introduced. These placeholder values can be zero for the log ID, expressing that no log entry occurred. For the time since log occurrence, it can be one, when the time since occurrence is expressed as a number between zero and one – zero standing for right now and one stands for before or at the beginning of the designated timeframe. When decoding the still active state, it can be zero for not active and for the segment, the no-location-available value is used. A graphical representation of the data quantities is shown in Fig. 5.

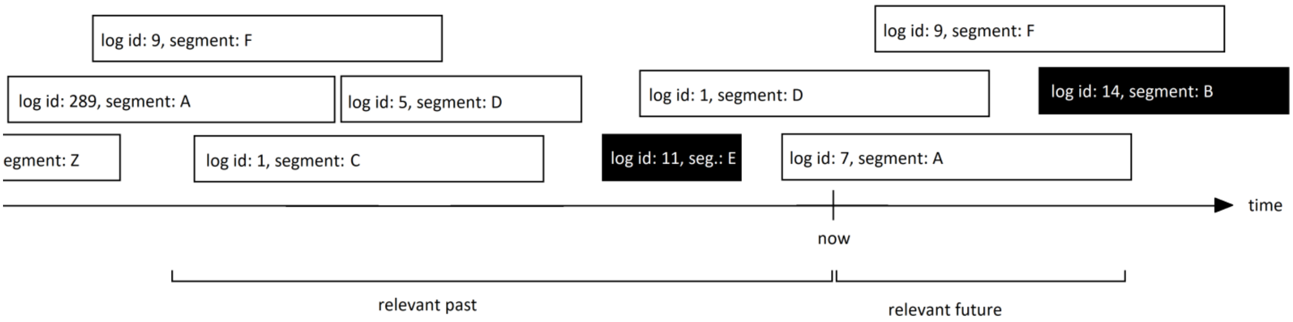


Fig. 5. Visualization of the AHT data. It contains a start and an end timestamp, a log ID, and a segment. Log IDs of interest are shown in a black rectangle. By introducing a relevant past, a relevant future, one can create input vectors for NNs.

Comprehensive NN VS. Multiple NNs

The theoretical advantages of using multiple NNs, one for each log ID of interest, were already discussed at the end of section III and can be extended by practical advantages. Besides the ability to gradually adapt the overall model to the concept drift and new log IDs, the possibility to create a well-performing overall model seems more easily achievable. Although a comprehensive NN predicting all log IDs of interest is theoretically equivalent to a combination of multiple NNs predicting a certain log ID, the computational effort can be reduced by following a divide-and-conquer approach and splitting the comprehensive NN into multiple NNs. This hard to quantify assumption was also observed during our experimentation.

We created a comparison between a comprehensive NN to multiple NNs in our production line setting. Therefore, we aimed to predict log IDs of special interest (based on domain experts' rating) which occur with a relative frequency of at least 0.11%. Furthermore, we used a weighted version of the cross-entropy loss [36] to account for the unbalanced nature of the data, where the weights are inversely proportional to the relative frequency of the corresponding log ID. The used architecture was the same for both cases and is shown in Fig. 6. The features were constructed from the last 100 log entries within the last 45 minutes, the labels were created based on the next 15 minutes. Each embedding layer is of dimension 8, each hidden dense layer consists out of 32 neurons, the drop layer features a dropout rate of 50%.

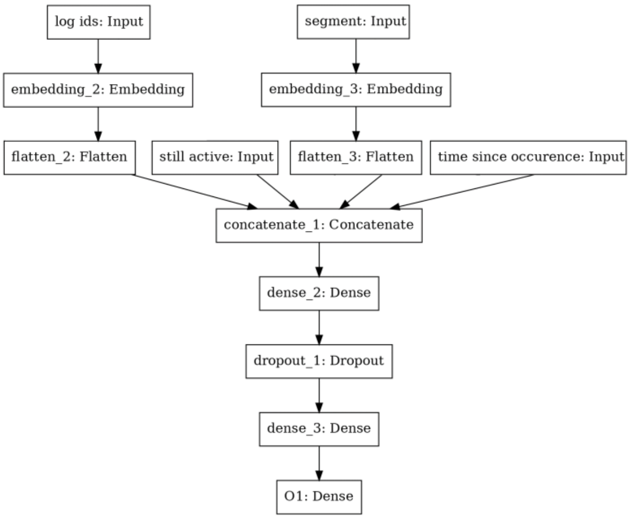


Fig. 6. NN architecture used for comparing a comprehensive NN to multiple NNs.

To compare the performance of the NNs, two widely used performance indicators are introduced – the positive predictive value (PPV) and the sensitivity. Both indicators are empirical probabilities. The indicator PPV can be calculated as the number of correct predictions of a log ID divided by the total number of predictions of this ID. The higher this value is, the more reliable is a gained prediction of this log ID. The indicator sensitivity can be calculated as the number of correct predictions of a log ID divided by the total number of occurrences of this log ID. The higher this indicator is, the fewer occurrences of this log ID are ‘overlooked’ by the NN. A reliable prediction system requires both values to be high.

Comparing the performance of a comprehensive NN to multiple NNs in our production line setting speaks for the usage of multiple NNs over a comprehensive NN. Table I shows the two introduced performance indicators, observed when predicted and predicted when observed, for both discussed cases. Except for log ID 4 and 8, both indicators speak consistently for using multiple NNs over a comprehensive NN. Although the log IDs 4 and 8 lead to contradicting indicators to some extent, the overall results speak clearly for using multiple NNs over a comprehensive NN.

TABLE I
PERFORMANCE COMPARISON BETWEEN A COMPREHENSIVE NN (COMNN) AND MULTIPLE NNS (MNNS) IN THE DESCRIBED PRODUCTION LINE SETTING WITH PERCENTAGE VALUES.

LOG ID	PPV		Sensitivity	
	ComNN	MNNs	ComNN	MNNs
1	79	**92**	1	**63**
2	34	**92**	46	**70**
3	65	**71**	32	**78**
4	58	**79**	**68**	66
5	54	**97**	13	**95**
6	47	**80**	50	**93**
7	71	**76**	16	**39**
8	**70**	44	68	**89**
9	65	**70**	8	**32**

The superior practical performance of multiple NNs over a comprehensive NN might be based on different circumstances. One such circumstance is that, instead of working with only one set of hyperparameters, each NN predicting only one log ID of interest allows for its own set of hyperparameters, such as relevant past (see Fig. 5) or the number of neurons in a certain layer. Another contributing factor might be the more complex structure of the comprehensive NN, which might induce a worse performance of the optimization algorithm used for the training procedure, leading to a suboptimal trained NN. Another factor is the design of the used labels, which allow for only one next log ID of interest to be predicted. This might distort the NN’s performance, since one log ID of interest might be concealed by another log ID of interest, leading to inferior performance.

To set the performance of the NNs from the multiple NNs approach into relation, one can compare it to trivial prediction models. A suitable base model can be obtained by always predicting the most common observed class. Due to the extremely unbalanced situation, we are facing with the AHT dataset, this is always clearly the prediction, that there will not be an ID of interest in the upcoming timeframe. The accuracy is calculated as the ratio of correct predictions to all predictions. For the no information rate model, this value is always the ratio of no-error predictions, which ranges for our situation between 89% to 98.6%. Although these values are quite high, the obtained NNs outperform this bae-line accuracy in nearly all cases with accuracies between 94.4% and 99.9%, as shown in Table II.

TABLE II
ACCURACY OF THE INDIVIDUAL MODELS FROM THE MULTIPLE NNS APPROACH COMPARED TO THE NO-INFORMATION-RATE MODEL (NEVER PREDICTING AN UPCOMING ID).

LOG ID	Accuracy	
	No information rate	MNNs
1	96.5	**98.6**
2	96.2	**98.6**
3	95.9	**97.8**
4	89.0	**94.2**
5	98.6	**99.9**
6	97.7	**99.3**
7	93.0	**94.9**
8	**95.4**	94.4
9	96.1	**96.9**

Proposed ALFA Software Design

To benefit from the advantages promoted in the previous sections, we propose the ALFA (AutoML for Log File Analysis) software design capable of handling the needed requirements. The software design contains two main components, the predictor and the model updater.

The predictor receives the log information as soon as they occur and creates a prediction based on this information. In the first step, the data – log ID, time, segment, and type (it is either the start or the end of a log event) is received and stored in a database. In a second step, the NNs are loaded from the model updater if they have not been loaded yet and enough data is available to do so. Finally, the NNs are used to predict the occurrence of log IDs of interest if enough data is available. Furthermore, the received log IDs can be used to calculate the performance of the previously made predictions, which guarantees for always present performance indicators for each NN. This component operates only, and as soon as new log data is received.

The model updater is used to store, load, and refit the NNs. It operates on an external trigger, either from the predictor when loading a NN, or a regular impulse, based on e.g., a certain time or a certain amount of received and relevant log data, to refit one or more NNs. When loading the NNs, the model updater first tries to load already existing NNs from the file system. If this is not possible, it loads the relevant log data to create and fit new NNs. In case that there is not enough data to fit NNs, nothing is done – until enough log data is present. Furthermore, on a regular basis, triggered e.g., by elapsed time or a certain amount of received data, the currently used NNs are reevaluated and possibly replaced, in other words, refitted. This includes loading of the relevant log data, the creation and fitting of new NNs, the comparison between these new NNs and the currently active NNs, and replacing the currently active NNs through the newly created and better performing NNs.

The software description is still lacking an essential piece of information, that is, what exactly we mean by creating and fitting a model. Recapitulating the discussions about NAS from Section II, we make use of architecture optimization in case of a neural network search space with subsequent hyperparameter optimization. The precedes tasks of data preparation and feature engineering, which are usually included into the AutoML pipeline are not required in the presented setting, since the data is already well formatted, and features have been predefined. The mode evaluation, on the other hand, is still a crucial aspect of the presented workflow and is used whenever a new NN is created and trained with the data. A schematic depiction of the described ALFA software architecture can be found in Fig. 7.

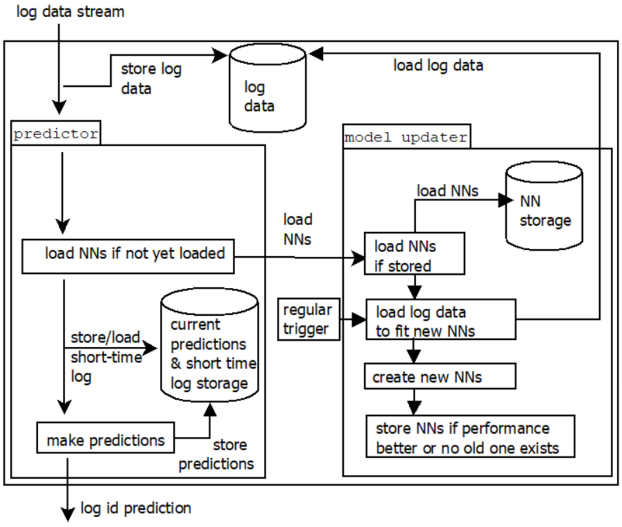


Fig. 7. ALFA software architecture proposed for the AHT use case, consisting of two main components, predictor and model updater.

This proposed software architecture is currently developed in the AHT project for industrial applications, especially for automated decision making by predictive diagnosis and machine learning in a semiconductor use case. The application is constantly enhanced. The search space includes the NN architecture shown in Fig. 6 and slight variations of it, the hyperparameter optimization uses a grid search approach. Possible extensions of this first implementation include the additional prediction of the occurrence segment and the extension of the currently used search space.

V. Conclusion

We have given an overview of the current state regarding log analysis and AutoML, furthermore, the advantages of combining these approaches were presented in theory and for the introduced AHT use case. In this context, the theoretical and practical predominance of using multiple NNs, each tuned to predict one log ID, over one comprehensive NN, predicting all log IDs, was shown. The invoked theoretical advantages are the possibility to gradually adapt the overall prediction model to the concept drift or even to a new quality of data, which are new log IDs in the AHT use case. The invoked practical advantage in the context at hand is the better results produced by using multiple NNs in the AHT use case, shown in Table I.

Based on these considerations, the ALFA software architecture comprised of multiple NNs was proposed, it is shown in Fig. 7. This architecture allows for the beforehand enumerated advantages and, beyond that, to also carry along the up-to-date performance of the used NNs. An automated update of the used NNs is done in the background, as soon as a better performing NN for a given log ID or even a new NN for a new log ID is found, it gets integrated. An expert in the field of NNs is not required to use this setup, opening the usage of it for a wide audience. The sole requirement is a suitable data format.

The gained models, from a first, simple implementation show great potential. Compared to the no-information-rate model, always predicting the most likely class, good performance was achieved. On the very unbalanced AHT dataset, assessed by the accuracy, the obtained NNs outperform the no-information-rate model in nearly all cases on a very high level, as shown in Table II.

The ALFA software architecture is currently developed in the AHT project and is constantly evaluated in this context as a new tool that will be provided to the Eclipse Arrowhead project. Although the first results are promising, a long-time evaluation might hold crucial information for the further development of the proposed prediction model. This especially concerns the evaluate the software component's behavior when confronted with an unknown concept drift and the introduction of data of new quality. This, however, requires time for data in the given setup to be shifted in this direction.

A next step, besides the long-term evaluation, is the extension of the ALFA software architecture. There are two apparent extensions, the introduction of an additional label dimension with the segment of occurrence, additionally to the log ID, and the improvement of the used NAS approach. The used NAS approach can be enhanced by extending the search space to include more NN architectures and by refining the hyperparameter optimization.

References

[1] Bao, L., Li, Q., Lu, P., Lu, J., Ruan, T., & Zhang, K. (2018). Execution anomaly detection in large-scale systems through console log analysis. Journal of Systems and Software, 143, 172-186. doi: 10.1016/j.jss.2018.05.016

[2] Boardman, J., & Sauser, B. (2006, April). System of Systems-the meaning of of. In 2006 IEEE/SMC International Conference on System of Systems Engineering (pp. 6-pp). IEEE. doi: 10.1109/SYSOSE.2006.1652284

[3] Carleo, G., Cirac, I., Cranmer, K., Daudet, L., Schuld, M., Tishby, N., ... & Zdeborová, L. (2019). Machine learning and the physical sciences. Reviews of Modern Physics, 91(4), 045002. doi: 10.1103/RevModPhys.91.045002

[4] Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R. D. P., Basto, J. P., & Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering, 137, 106024. doi: 10.1016/j.cie.2019.106024

[5] Chen, J., Chen, K., Chen, X., Qiu, X., & Huang, X. (2018). Exploring shared structures and hierarchies for multiple nlp tasks. arXiv preprint arXiv:1808.07658.

[6] Chen, W., Wilson, J., Tyree, S., Weinberger, K., & Chen, Y. (2015, June). Compressing neural networks with the hashing trick. In International conference on machine learning (pp. 2285-2294). PMLR.

[7] Domhan, T., Springenberg, J. T., & Hutter, F. (2015, June). Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves. In Twenty-fourth international joint conference on artificial intelligence.

[8] Dikov, G., & Bayer, J. (2019, April). Bayesian learning of neural network architectures. In The 22nd International Conference on Artificial Intelligence and Statistics (pp. 730-738). PMLR.

[9] Eggensperger, K., Hutter, F., Hoos, H., & Leyton-Brown, K. (2015, February). Efficient benchmarking of hyperparameter optimizers via model-based surrogates. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 29, No. 1). DOI: 10.1007/s10994-017-5683-z

[10] Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. ACM computing surveys (CSUR), 46(4), 1-37. DOI: 10.1145/2523813

[11] Ghiasi, G., Lin, T. Y., & Le, Q. V. (2019). Nas-fpn: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 7036-7045). DOI: 10.1109/cvpr.2019.00720

[12] He, S., Lin, Q., Lou, J. G., Zhang, H., Lyu, M. R., & Zhang, D. (2018, October). Identifying impactful service system problems via log analysis. In Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering (pp. 60-70). DOI: 10.1145/3236024.3236083

[13] He, S., Zhu, J., He, P., & Lyu, M. R. (2016, October). Experience report: System log analysis for anomaly detection. In 2016 IEEE 27th International Symposium on Software Reliability Engineering (ISSRE) (pp. 207-218). IEEE. DOI: 10.1109/ISSRE.2016.21

[14] He, X., Zhao, K., & Chu, X. (2021). AutoML: A Survey of the State-of-the-Art. Knowledge-Based Systems, 212, 106622. DOI: 10.1016/j.knosys.2020.106622

[15] Jayathilake, D. (2012, May). Towards structured log analysis. In 2012 Ninth International Conference on Computer Science and Software Engineering (JCSSE) (pp. 259-264). IEEE. DOI: 10.1109/jcsse.2012.6261962

[16] Jiang, Y., Hu, C., Xiao, T., Zhang, C., & Zhu, J. (2019, November). Improved differentiable architecture search for language modeling and named entity recognition. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 3576-3581). DOI: 10.18653/v1/D19-1367

[17] Kandasamy, K., Neiswanger, W., Schneider, J., Poczos, B., & Xing, E. (2018). Neural architecture search with bayesian optimisation and optimal transport. arXiv preprint arXiv:1802.07191.

[18] Klein, A., Falkner, S., Bartels, S., Hennig, P., & Hutter, F. (2017, April). Fast bayesian optimization of machine learning hyperparameters on large datasets. In Artificial Intelligence and Statistics (pp. 528-536). PMLR.

[19] Lee, J. H., Shin, J., & Realff, M. J. (2018). Machine learning: Overview of the recent progresses and implications for the process systems engineering field. Computers & Chemical Engineering, 114, 111-121. DOI: 10.1016/j.compchemeng.2017.10.008

[20] Liu, C., Chen, L. C., Schroff, F., Adam, H., Hua, W., Yuille, A. L., & Fei-Fei, L. (2019). Auto-deeplab: Hierarchical neural architecture search for semantic image segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 82-92). DOI: 10.1109/CVPR.2019.00017

[21] Liu, C., Zoph, B., Neumann, M., Shlens, J., Hua, W., Li, L. J., ... & Murphy, K. (2018). Progressive neural architecture search. In Proceedings of the European conference on computer vision (ECCV) (pp. 19-34). DOI: 10.1007/978-3-030-01246-5_2

[22] Liu, H., Simonyan, K., Vinyals, O., Fernando, C., & Kavukcuoglu, K. (2017). Hierarchical representations for efficient architecture search. arXiv preprint arXiv:1711.00436.

[23] Liu, H., Simonyan, K., & Y ang, Y . (2018). Darts: Differentiable architecture search. arXiv preprint arXiv:1806.09055.

[24] Lin, T. T., & Siewiorek, D. P. (1990). Error log analysis: statistical modeling and heuristic trend analysis. IEEE Transactions on reliability, 39(4), 419-432. DOI: 10.1016/0026-2714(92)90140-g

[25] Maier, M. W. (1998). Architecting principles for systems-of-systems. Systems Engineering: The Journal of the International Council on Systems Engineering, 1(4), 267-284. DOI: 10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D

[26] Mendoza, H., Klein, A., Feurer, M., Springenberg, J. T., & Hutter, F. (2016, December). Towards automatically-tuned neural networks. In Workshop on Automatic Machine Learning (pp. 58-65). PMLR. DOI: 10.1007/978-3-030-05318-5_7

[27] Mobley, R. K. (2002). An introduction to predictive maintenance. Elsevier. DOI: 10.1016/b978-0-7506-7531-4.x5000-3

[28] Oliner, A., Ganapathi, A., & Xu, W. (2012). Advances and challenges in log analysis. Communications of the ACM, 55(2), 55-61. DOI: 10.1145/2076796.2082137

[29] Pham, H., Guan, M., Zoph, B., Le, Q., & Dean, J. (2018, July). Efficient neural architecture search via parameters sharing. In International Conference on Machine Learning (pp. 4095-4104). PMLR.

[30] Selcuk, S. (2017). Predictive maintenance, its implementation and latest trends. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 231(9), 1670-1679. DOI: 10.1177/0954405415601640

[31] Sipos, R., Fradkin, D., Moerchen, F., & Wang, Z. (2014, August). Log-based predictive maintenance. In Proceedings of the 20th ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1867-1876). DOI: 10.1145/2623330.2623340

[32] Tan, W. N. (2019). SMT Machine Log File PDE Features Extraction and Analysis (Doctoral dissertation, Tunku Abdul Rahman University College).

[33] Wei, T., Wang, C., Rui, Y., & Chen, C. W. (2016, June). Network morphism. In International Conference on Machine Learning (pp. 564- 572). PMLR.

[34] Zela, A., Klein, A., Falkner, S., & Hutter, F. (2018). Towards automated deep learning: Efficient joint neural architecture and hyperparameter search. arXiv preprint arXiv:1807.06906.

[35] Zhang, H., Li, Y., Chen, H., & Shen, C. (2019). Ir-nas: Neural architecture search for image restoration. arXiv preprint arXiv:1909.08228.

[36] Zhang, Z., & Sabuncu, M. R. (2018). Generalized cross entropy loss for training deep neural networks with noisy labels. arXiv preprint arXiv:1805.07836.

[37] Zoph, B., & Le, Q. V. (2016). Neural architecture search with reinforcement learning. arXiv preprint arXiv:1611.01578.

Matthias Maurer is Senior Researcher for Contextual Information Systems and Operational Insights at Virtual Vehicle Research GmbH. His research interests include data analytics, machine learning, and cognitive science. Matthias has received his Dipl.-Ing. (equiv. MSc) from Graz University of Technology in Technical Mathematics and his Mag. rer. nat. from University of Graz in Education. His work is manly focused on the analysis of time-based production measurement and log data, focusing on predictive maintenance and root cause analysis..

Dipl.-Ing. **Andreas Festl**, BSc, is Senior Researcher for Contextual Information Systems and Management at Virtual Vehicle Research GmbH and affiliated lecturer for data and information science at FH Joanneum University of applied sciences. His research interests include data analytics, statistics and machine learning. Andreas has received his Dipl.-Ing. (equiv. MSc) from Graz University of Technology in Technical Mathematics. A large part of his work was and is focused on the analysis of time-based automotive measurement data, thereby answering questions about customer vehicle usage, driving behavior and various environmental conditions.

Mag. Bor Bricelj, CQRM holds a master's degree in economics and finance from University of Maribor’s Faculty of Economics and Business, and a Certificate in Quantitative Risk Management from the International Institute of Professional Education and Research. He worked in the fields of financial services and higher education before transitioning to data science. He has more than five years of experience as a data scientist, implementing statistical and machine learning methods to analyse and solve various industry specific problems in different industry branches, ranging from automotive industry, heavy industry, to chemical industry. At Virtual Vehicle Research GmbH, he is employed as a Senior researcher / Data Scientist, working with the "Information Network Extraction Systems" group. His work and research are focused on domains of computer vision and data enrichment.

Dr. Germar Schneider (m) holds a Diploma and a PhD in chemistry. He joined the Siemens AG in Essonnes in France in 1995 as a process engineer in the wet department. In 1998 in Dresden, he became the section manager for the 200mm wet department. From 2004 to 2008, he built up a team that was important for factory automation. Between 2008 and 2012, as manager in the new wafer test department he was responsible for production & maintenance and equipment engineering. With 26 years of experience combining know-how of process engineering, production, maintenance, automation and the experience in digitalization projects he is main driver in various JU projects.

Dr. Michael Schmeja (55) has been working at the Virtual Vehicle Research Center in Graz since 2009 and is currently the responsible Area Manager for Safety & Security. After studying mathematics, Mr. Schmeja earned his doctorate at the Institute of Railway Engineering at the Graz University of Technology. From 1997 - 2009 he held various management positions at Siemens Mobility, where he was awarded Inventor of the Year in 2003. In 2009, he moved to Virtual Vehicle and, in addition to his function as Area Manager, he also manages numerous international research projects.

IEEE/IFIP Network Operations and Management Symposium

25-29 April 2022 // Budapest, Hungary

Network and Service Management in the Era of Cloudification, Softwarization and Artificial Intelligence

Doctoral Symposium

IEEE ComSoc IEEE Communications Society | ifip | IEEE

CALL FOR DOCTORAL SYMPOSIUM

OVERVIEW

The Doctoral Symposium is a concentrated half-day event that will be held during IEEE/IFIP NOMS 2022, which takes place in Budapest between 25 and 29 April 2022. In it, PhD students present their ongoing research interests/plans/results to a panel of researchers in the field and receive specific constructive feedback, including opportunities to meet with mentors one-on-one. Additionally, accepted students will create a poster about their work to allow IEEE/IFIP NOMS attendees to quickly familiarize themselves with each other's work and to be shown to a broader audience during the conference. Accepted position papers (four pages maximum) will be published in the conference proceedings and to be included in the online IEEE/IFIP NOMS conference proceedings. Moreover, the abstracts will be archived in the IEEExplore digital library once presented by the PhD students during the conference.

ELIGIBILITY AND TOPICS

Ideal candidates will be in the early or mid stages of their PhD and should have a solid idea of their direction and topic, but should also have room for improvement. The Doctoral Symposium welcomes applicants from a broad range of disciplines including Management of 6G Networks and Network 2030, Management of Smart Vertical Systems in the Industry 4.0 Era, Artificial Intelligence techniques for Network and Service Management, Management of Softwarized Networks, Software-Defined Networking, Network Function Virtualization, Service Function Chaining, VR, Service and application management, and related fields.

SUBMISSION GUIDELINES

To apply to the Doctoral Symposium, please submit the following two documents through JEMS at: *https://jems.sbc.org.br/noms2022*

- Position paper: Paper submissions should be 4 pages in length including references, using the IEEE 2-column. This position paper must clearly motivate, discuss, and summarize the proposed Ph.D. research, describe how the research fits into and advances research in related fields, and report on your progress. You may additionally focus on a more specific area of the research if desired. In the concluding section, you should identify 2 questions/areas for improvement that you would like to discuss during the session.
- A one page Curriculum Vitae.

IMPORTANT DATES

Submission Deadline: 30 November 2021
Notification of Acceptance: 20 December 2021
Camera-Ready: 14 January 2022

CHAIRS

Maria Torres Vega (Ghent University - imec, Belgium)

For more information, please visit http://noms2022.ieee-noms.org

IEEE MELECON 2022

The 21st IEEE Mediterranean Electrotechnical Conference

Palermo, Italy, 14-16 June 2022

Organized by the IEEE Italy Section and University of Palermo - https://www.melecon2022.org

CALL FOR PAPERS

IEEE MELECON 2022 is a major international forum presenting design methodologies, techniques, and experimental results in emerging electro-technologies. It is one of the flagship conferences of the IEEE Region 8 (the largest region of IEEE including Europe, Africa, and Middle East). It is expected to bring together researchers and practitioners from different fields of Electrical Engineering. The technical program will include plenary sessions, regular technical sessions, special sessions, panels, tutorials, and special events devoted to students and young professionals, Women in Engineering, entrepreneurs and industries. For this edition, the technical sessions are organized into four main tracks.

TRACK 1: SMART ENERGY

Chairs

Carlo Cecati, IES Italy Chapter Chair
Fabio Viola, University of Palermo
Hadi Kanaan, Saint Joseph Univ. Beirut, Lebanon

1.1. *Conversion and Control of Sustainable Energy Sources*
1.2. *Power Electronics and Control in Smart Grids, Industry and e-Transportation*
1.3. *Energy Storage Systems and their Control*
1.4. *Electrical Machines and Drives for Industry and Renewable energy Systems*
1.5. *Energy Management, Smart Metering and Distributed Energy Resources*
1.6. *Electric Mobility: challenges, trends, safety and EMI issues*
1.7. *Application of Machine Learning and Artificial Intelligence in Smart Grids*
1.8. *Energy Harvesting, Wireless Power transfer and Power Electronics Systems*
1.9. *Cyber Security and Big Data Issues for Smart Grid Systems*

TRACK 2: SMART INDUSTRY

Chairs

Giambattista Gruosso, Politecnico di Milano
Federico Baronti, University of Pisa
………..

2.1. *Smart Materials & Smart Sensor for Industry 4.0*
2.2. *Sustainable Industrial Processes and Products*
2.3. *Modelling and Simulation of Advanced Products and Manufacturing Processes*
2.4. *Additive Manufacturing Technologies, Applications and Measurement*
2.5. *Smart Systems and Artificial Intelligence for Manufacturing*
2.6. *Smart Technology for Autonomous Systems: from Robots to Drone*
2.7. *Collaborative Machines and Systems*
2.8. *Augmented Reality-Based and Virtual Reality*
2.9. *Agriculture 4.0: Technology and Solution*
2.10. *Digital Manufacturing: from Data Collection to Intelligent Systems*

TRACK 3: SMART HEALTHCARE

Chairs

Sergio Cerutti, EMB Italy Chapter Chair
Thomas Penzel, Charité University Hospital, Berlin, Germany

3.1. *Services, Applications and Solutions in Smart Healthcare*
3.2. *Big Data Integration and Personalised Medicine*
3.3. *E-Health and IoT for Smart HealthCare*
3.4. *Neural and Cognitive Engineering*
3.5. *Advances in Medical Informatics for HealthCare Applications*
3.6. *Biotechnologies: advanced Devices and Sensors*
3.7. *Bio-electromagnetic modelling*
3.8. *Nanostructured devices and smart materials for biophotonics applications*

TRACK 4: SMART DIGITAL COMMUNITIES

Chairs

Barbara Masini, CNR-IEIIT
Francesco Masulli, CI Italy Chapter Chair
Alexey Vinel, Halmstad University, Sweden

4.1. *Smart Education Technologies*
4.2. *Cognitive Computing, Artificial Intelligence & Machine Learning*
4.3. *Semantic Web, Big data & Analytics*
4.4. *Smart Living Technologies*
4.5. *5G and beyond Wireless Networks*
4.6. *Digital Twins for Smart Cities*
4.7. *3D Networks (terrestrial and aerial)*
4.8. *Remote Sensing Methods and Applications*
4.9. *IoT and Smart Communications*
4.10. *Smart Mobility and Transportation*

Honorary Chair
Antonio Luque, IEEE R8 Director

General Chairs
Guido Ala, University of Palermo
Sergio Rapuano, IEEE Italy Section Chair
Tiziana Tambosso, IEEE R8 CoCSC Representative

Steering Committee Chairs
Bernardo Tellini, IEEE Italy Section Past-Chair
Ermanno Cardelli, IEEE Italy Section National Association Liaison Committee Coordinator

National research associations
Riccardo Leonardi, Research Association GTTI
Ermanno Cardelli, Research Association ET
Paolo Carbone, Research Association GMEE
Alfonso Damiano, Research Association CMAEL
Emilio Ferrari, Research Association AIDI
Nicola Paone, Research Association GMMT
Giuseppe Mazzarella, Research Association SIEM
Dario Zaninelli, Research Association GUSEE
Paolo Atzeni , Research Association GII
Giovanni Ghione, Research Association SIE
Sauro Longhi, Research Association SIDRA
Luca Formaggia, SIMAI

International Steering Committee
Maddalena Salazar Palma, Past-Director Region 8
Vincenzo Piuri, Director-Elect Region 8
Habib Kammoun, R8 CoCSC Chair
Mona Gassemian, UK&Ireland Session Chair
Shmuel Auster, Israel Section Chair
Chiara Boccaletti, IAS Meeting Department Chair
EMBS Representative
Mohammed El Mohajir, MELECON 2018 General Chair

Technical Program Committee Chair
Gianfranco Chicco, IEEE Italy Section Vice-Chair

Technical Program Committee co-Chair
Daniela Proto, IEEE Italy Section Conference Committee Coordinator

Publication Chairs
Pietro Romano, University of Palermo
Gaetano Zizzo, University of Palermo

Professional Activities
Stefano Massucco, IEEE Italy Section Professional Activity Committee Coordinator
Federica Battisti, IEEE Italy Section YP-AG Chair

Publicity Chair
Stefano Ferrari, Italy Section Membership Development Committee Coordinator
Salvatore Favuzza, Elisa Francomano, Vincenzo Di Dio, University of Palermo

Special Meeting and Exhibition for "innovative Start up and Entrepreneurship"
Tiziana Tambosso, IEEE Italy Section Entrepreneurship Committee Coordinator
Vincenzo Piuri, IEEE R8 Director-Elect
Marios Antoniou, IEEE R8 Entrepreneurship Initiative

Women In Engineering Event
Dajana Cassioli, IEEE Italy Section WIE-AG Chair
Patrizia Lamberti, IEEE Italy Section WIE-AG Vice-Chair
R8 WIE SC Representative

Student and Young Professional Events
Paolo Maresca, IEEE Italy Section SAC Coordinator
Federica Battisti, IEEE Italy Section YP-AG Chair
Gaetano Zizzo, University of Palermo
R8 SAC Representative
R8 YP SC Representative

Special Meeting with Industry
Dario Petri, Italy Section Industry Relation Committee Coordinator
John Matogo R8 Action for Industry Chair

Tutorial Chair
Rossano Musca, University of Palermo

Treasurer
Pisana Placidi, IEEE Italy Section Treasurer, University of Perugia

Local Organizing Committee
Guido Ala, Gaetano Zizzo, Salvatore Favuzza, Fabio Viola,
Giuseppe Rizzo, Pietro Romano, Antonino Imburgia, Giuseppe Schettino

Secretariat
Fabio Viola, University of Palermo

Webmaster
Gianluca Mazzilli, Athena srl
Prospective

Prospective Authors of papers are invited to submit a paper (typically 4-6 pages in standard IEEE two-column format) via EDAS by suggesting the related Track and Technical Session. The paper should contain a complete description of the proposed contribution along with results, suitably framed in the related state of the art. Each paper will be reviewed in terms of relevance with respect to the scope of the event, originality and quality of the technical content, overall organization and writing style. Papers must be prepared according to the Author's instructions reported on the MELCON2022 website: **https://www.melecon2022.org**

Submission of papers implies intention to register and present the related content at the conference. Proceedings papers presented at the Conference will be submitted for inclusion in the IEEE Xplore Digital Library.

IMPORTANT DATES

Deadline for Submission of Special Session Proposals	**17 December 2021**
Deadline for Submission of Papers	**22 January 2022**
Deadline for Submission of Tutorials	**18 February 2022**
Notification of Acceptance	**11 March 2022**
Deadline for Submission of Camera-Ready Papers	**8 April 2022**
Early Registration	**18 April 2022**

Extended version of papers presented at the conference, which fall within the scope of the IEEE Industry Application Society (Technical Sponsor of IEEE MELECON 2022) may be submitted to the IEEE Transaction on Industry Application for publication. IEEE Transactions on Industry Applications is the peer-reviewed IAS 'journal of record' and all papers presented in IAS Transactions have previously been presented at a technical conference.

Extended version of the best papers presented at the conference, which fall within the scope of the IEEE Journal of Translational Engineering in Health and Medicine may be submitted for publication to the JTEHM.

Special Issue

of the Infocommunication Journal

Call for Papers

Tech-Augmented Legal Environment

The scope of the special issue is to focus on the relevant changes in law and legal institutions due to the contemporary technological and digital innovations. It has been previously established in academic literature that digital technologies became part of the everyday reality, interacting with and constantly forming the environment of humans. The fundamental modification of the usage and intervention of tech-based tools in humans daily practice must be transformed to a modified view on the regulatory role, as well as academic research. The lectures shall pay an in-depth attention to digitalization and its effect on the general legal system with special regard (but not exclusively) to civil law, administrative law, theory of law, legal ethics, European and international law. The papers shall underline the most relevant present challenges of digitalization on the entire legal system.

Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

ISSN 2061-2079

Special Issue

Technically Co-Sponsored by

IEEE ComSoc IEEE Communications Society

hte

IEEE HUNGARY SECTION

This special issue collects the latest results emerging on the field of Tech-Augmented Legal Environment.

Guest Editors:

Dr. Gábor Kecskés
Széchenyi István University (Hungary)
Dr. Rastislav Funta
Danubius University (Slovakia)

Important dates:

Submission paper deadline: **1st of February, 2022**
Notification first review: **1st of March, 2022**
Deadline for revised paper: **10th of April, 2022**
Camera Ready: **20th of April, 2022**

Guidelines for our Authors

Format of the manuscripts

Original manuscripts and final versions of papers should be submitted in IEEE format according to the formatting instructions available on

https://journals.ieeeauthorcenter.ieee.org/
Then click: "IEEE Author Tools for Journals"
- "Article Templates"
- "Templates for Transactions".

Length of the manuscripts

The length of papers in the aforementioned format should be 6-8 journal pages.
Wherever appropriate, include 1-2 figures or tables per journal page.

Paper structure

Papers should follow the standard structure, consisting of *Introduction* (the part of paper numbered by "1"), and *Conclusion* (the last numbered part) and several *Sections* in between.
The Introduction should introduce the topic, tell why the subject of the paper is important, summarize the state of the art with references to existing works and underline the main innovative results of the paper. The Introduction should conclude with outlining the structure of the paper.

Accompanying parts

Papers should be accompanied by an *Abstract* and a few *Index Terms (Keywords)*. For the final version of accepted papers, please send the short cvs and *photos* of the authors as well.

Authors

In the title of the paper, authors are listed in the order given in the submitted manuscript. Their full affiliations and e-mail addresses will be given in a footnote on the first page as shown in the template. No degrees or other titles of the authors are given. Memberships of IEEE, HTE and other professional societies will be indicated so please supply this information. When submitting the manuscript, one of the authors should be indicated as corresponding author providing his/her postal address, fax number and telephone number for eventual correspondence and communication with the Editorial Board.

References

References should be listed at the end of the paper in the IEEE format, see below:

a) Last name of author or authors and first name or initials, or name of organization
b) Title of article in quotation marks
c) Title of periodical in full and set in italics
d) Volume, number, and, if available, part
e) First and last pages of article
f) Date of issue
g) Document Object Identifier (DOI)

[11] Boggs, S.A. and Fujimoto, N., "Techniques and instrumentation for measurement of transients in gas-insulated switchgear," IEEE Transactions on Electrical Installation, vol. ET-19, no. 2, pp.87–92, April 1984. DOI: 10.1109/TEI.1984.298778

Format of a book reference:

[26] Peck, R.B., Hanson, W.E., and Thornburn, T.H., Foundation Engineering, 2nd ed. New York: McGraw-Hill, 1972, pp.230–292.

All references should be referred by the corresponding numbers in the text.

Figures

Figures should be black-and-white, clear, and drawn by the authors. Do not use figures or pictures downloaded from the Internet. Figures and pictures should be submitted also as separate files. Captions are obligatory. Within the text, references should be made by figure numbers, e.g. "see Fig. 2."
When using figures from other printed materials, exact references and note on copyright should be included. Obtaining the copyright is the responsibility of authors.

Contact address

Authors are requested to submit their papers electronically via the following portal address:

https://www.ojs.hte.hu/infocommunications_journal/about/submissions

If you have any question about the journal or the submission process, please do not hesitate to contact us via e-mail:

Editor-in-Chief: Pál Varga – pvarga@tmit.bme.hu

Associate Editor-in-Chief:

Rolland Vida – vida@tmit.bme.hu

László Bacsárdi – bacsardi@hit.bme.hu

Special Issue

of the **Infocommunication Journal**

Call for Papers

Internet of Digital Reality: Applications and Key Challenges

A Digital Reality (DR) is a high-level integration of virtual reality (including augmented reality, virtual and digital simulations and twins), artificial intelligence and 2D digital environments which creates a highly contextual reality for humans in which previously disparate realms of human experience are brought together. DR encompasses not only industrial applications but also helps increase productivity in all corners of life (both physical and digital), thereby enabling the development of new social entities and structures, such as 3D digital universities, 3D businesses, 3D governance, 3D web-based digital entertainment, 3D collaborative sites and marketplaces. The Internet of Digital Reality (IoD) is a set of technologies that enables digital realities to be managed, transmitted and harmonized in networked environments (both public and private), focusing on a higher level of user accessibility, immersiveness and experience with the help of virtual reality and artificial intelligence.

Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

ISSN 2061-2079

Special Issue

Technically Co-Sponsored by IEEE ComSoc IEEE Communications Society

hte

IEEE HUNGARY SECTION

This special issue collects the latest results emerging on the field of Cognitive Infocommunications.

Chief Editor:
Prof. Peter Baranyi
Széchenyi István University

Guest Editors:
Prof. György Wersényi
Széchenyi István University
Dr. Ádám Csapó
Széchenyi István University
Prof. Anna Esposito
Università degli Studi della Campania "Luigi Vanvitelli"
Prof. Atsushi Ito
Utsunomiya University, Japan

Important dates:

Submission paper deadline: **1st of February, 2022**
Notification first review: **1st of March, 2022**
Deadline for revised paper: **10th of April, 2022**
Camera Ready: **20th of April, 2022**

Regarding manuscript submission information, please visit:
https://www.infocommunications.hu/for-our-authors

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

hte

FOUNDED IN 1949

SCIENTIFIC
ASSOCIATION FOR
INFOCOMMUNICATIONS
HUNGARY

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: Hírközlési és Informatikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society.

What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

Contact information

President: **FERENC VÁGUJHELYI** • *elnok@hte.hu*
Secretary-General: **ISTVÁN MARADI** • *istvan.maradi@gmail.com*
Operations Director: **PÉTER NAGY** • *nagy.peter@hte.hu*
International Affairs: **ROLLAND VIDA, PhD** • *vida@tmit.bme.hu*

Address: H-1051 Budapest, Bajcsy-Zsilinszky str. 12, HUNGARY, Room: 502
Phone: +36 1 353 1027
E-mail: *info@hte.hu*, Web: *www.hte.hu*