Annals of Computer Science and Information Systems
Volume 19

Position Papers of the 2019 Federated Conference on Computer Science and Information Systems

September 1–4, 2019. Leipzig, Germany

Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki (eds.)

PTI

Annals of Computer Science and Information Systems, Volume 19

Position Papers of the 2019 Federated Conference on Computer Science and Information Systems

Maria Ganzha, Leszek Maciaszek, Marcin Paprzycki
(eds.)

POLSKIE TOWARZYSTWO INFORMATYCZNE
POLISH INFORMATION PROCESSING SOCIETY

2019, Warszawa, Polskie
Towarzystwo Informatyczne

Annals of Computer Science and Information Systems, Volume 19
Position Papers of the 2019 Federated Conference on Computer Science and Information Systems

USB: ISBN 978-83-955416-2-9
WEB: ISBN 978-83-955416-1-2

ISSN 2300-5963
DOI 10.15439/978-83-955416-1-2

Ul. Solec 38/103
00-394 Warsaw
Poland

Contact: secretariat@fedcsis.org
`http://annals-csis.org/`

Cover art:
Edyta Paluszyńska,
Elbląg, Poland

Also in this series:

Volume 20: Communication Papers of the 2019 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-955416-3-6, ISBN USB: 978-83-955416-4-3**
Volume 18: Proceedings Papers of the 2019 Federated Conference on Computer Science and Information Systems, **ISBN Web 978-83-952357-8-8, ISBN USB 978-83-952357-9-5, ISBN ART 978-83-955416-0-5**
Volume 17: Communication Papers of the 2018 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-952357-0-2, ISBN USB: 978-83-952357-1-9**
Volume 16: Position Papers of the 2018 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-949419-8-7, ISBN USB: 978-83-949419-9-4**
Volume 15: Proceedings Papers of the 2018 Federated Conference on Computer Science and Information Systems, **ISBN Web 978-83-949419-5-6, ISBN USB 978-83-949419-6-3, ISBN ART 978-83-949419-7-0**
Volume 14: Proceedings of the First International Conference on Information Technology and Knowledge Management, **ISBN WEB: 978-83-949419-2-5, ISBN USB: 978-83-949419-1-8, ISBN ART: 978-83-949419-0-1**
Volume 13: Communication Papers of the 2017 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-922646-2-0, ISBN USB: 978-83-922646-3-7**
Volume 12: Position Papers of the 2017 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-922646-0-6, ISBN USB: 978-83-922646-1-3**
Volume 11: Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-946253-7-5, ISBN USB: 978-83-946253-8-2, ISBN ART: 978-83-946253-9-9**
Volume 10: Proceedings of the Second International Conference on Research in Intelligent and Computing in Engineering, **ISBN WEB: 978-83-65750-05-1, ISBN USB: 978-83-65750-06-8**
Volume 9: Position Papers of the 2016 Federated Conference on Computer Science and Information Systems, **ISBN WEB: 978-83-60810-93-4, ISBN USB: 978-83-60810-94-1**

DEAR Reader, it is our pleasure to present to you Position Papers of the 2019 Federated Conference on Computer Science and Information Systems (FedCSIS), which took place in Leipzig, Germany, on September 1-4, 2019.

Position papers comprise two categories of contributions – challenge papers and emerging research papers. *Challenge papers* propose and describe research challenges in theory or practice of computer science and information systems. Papers in this category are based on deep understanding of existing research or industrial problems. Based on such understanding and experience, they define new exciting research directions and show why these directions are crucial to the society at large. *Emerging research papers* present preliminary research results from work-in-progress based on sound scientific approach but presenting work not completely validated as yet. They describe precisely the research problem and its rationale. They also define the intended future work including the expected benefits from solution to the tackled problem. Subsequently, they may be more conceptual than experimental.

FedCSIS 2019 was Chaired by prof. Bogdan Franczyk, while prof. Rainer Unland acted as the Chair of the Organizing Committee. This year, FedCSIS was organized by the Polish Information Processing Society (Mazovia Chapter), IEEE Poland Section Computer Society Chapter, Systems Research Institute Polish Academy of Sciences, Warsaw University of Technology, Wrocław University of Economics, and Leipzig University, Germany.

FedCSIS 2019 was technically co-sponsored by: IEEE Region 8, IEEE Poland Section, IEEE Computer Society Technical Committee on Intelligent Informatics, IEEE Czechoslovakia Section Computer Society Chapter, IEEE Poland Section Gdańsk Computer Society Chapter, IEEE Poland Section Systems, Man, and Cybernetics Society Chapter, IEEE Poland Section Control System Society Chapter, IEEE Poland Section Computational Intelligence Society Chapter, Committee of Computer Science of the Polish Academy of Sciences, Polish Operational and Systems Research Society, Mazovia Cluster ICT Poland and Eastern Cluster ICT Poland. FedCSIS 2019 was sponsored by Intel.

During FedCSIS 2019, keynote lectures were delivered by:

- Enrique Alba, University of Málaga, Spain, "*Intelligent Systems for Smart Cities*"
- Francisco Herrera, Dept. Computer Sciences and Artificial Intelligence Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI) University of Granada, "*Deep Data and Big Learning: More quality data for better knowledge"*
- George Spanoudakis, Research Centre for Adaptive Computing Systems (CeNACS), School of Mathematics, Computer Science and Engineering, City, University of London, "*Cyber security risks: Comprehensive mitigation through technical, contractual and financial mitigation mechanisms"*

FedCSIS 2019 consisted of five Tracks and a doctoral symposium. Tracks were divided into Technical Sessions. Sessions were preannounced in Call for Papers as track-related events (conferences, symposia, workshops, special sessions).

- **Track 1: Artificial Intelligence and Applications**
  - Advances in Artificial Intelligence and Applications (14th Symposium AAIA'19)
  - Computational Optimization (12th Workshop WCO'19)
  - Smart Energy Networks & Multi-Agent Systems (7th Workshop SEN-MAS'19)
- **Track 2: Computer Science & Systems**
  - Computer Aspects of Numerical Algorithms (12th Workshop CANA'19)
  - Cryptography and Security Systems (6th Conference C&SS'19)
  - Language Technologies and Applications (4th Workshop LTA'19)
  - Multimedia Applications and Processing (12th Symposium MMAP'19)
  - Advances in Programming Languages (7th Workshop WAPL'19)
  - Scalable Computing (10th Workshop WSC'19)
- **Track 3: Network Systems and Applications**
  - Advances in Network Systems and Applications (ANSA)
  - Internet of Things - Enablers, Challenges and Applications (3rd Workshop IoT-ECAW'19)
- **Track 4: Information Systems and Technology**
  - Advanced Information Technologies for Management (16th Conference AITM'19)
  - Data Science in Health (1st Special Session DSH'19)
  - Data Analysis and Computation for Digital Ecosystems (1st Workshop InC2Eco'19)
  - Information Systems Management (14th Conference ISM'19)
  - Knowledge Acquisition and Management (25th Conference KAM'19)
- **Track 5: Software and System Engineering**
  - Advances in Software and System Engineering (ASSE)
  - Cyber-Physical Systems (6th Workshop IWCPS-6)
  - Lean and Agile Software Development (3rd International Conference LASD'19)
  - Multimedia, Interaction, Design and Innovation (7th Conference MIDI'19)
  - Software Engineering (39th IEEE Workshop SEW-39)

Each paper, found in this volume, was refereed by at least two referees.

The program of FedCSIS required a dedicated effort of many people. Each event constituting FedCSIS had its own Organizing and Program Committee. We would like to express our warmest gratitude to all Committee members for their hard work in attracting and later refereeing 298 regular submissions.

We thank the authors of papers for their great contribution to research and practice in computing and information systems. We thank the invited speakers for sharing their knowledge and wisdom with the participants. Finally, we thank all those responsible for staging the conference in Leipzig. Organizing a conference of this scope and level could only be achieved by the collaborative effort of a highly capable team taking charge of such matters as conference registration system, finances, the venue, social events, catering, handling all sorts of individual requests from the authors, preparing the conference rooms, etc.

We hope you had an inspiring conference and an unforgettable stay in the beautiful city of Leipzig. We also hope to meet you again for 2020 in Sofia, Bułgaria.

Co-Chairs of the FedCSIS Conference Series

Maria Ganzha, *Warsaw University of Technology, Poland and Systems Research Institute Polish Academy of Sciences, Warsaw, Poland*
Leszek Maciaszek, *Wrocław University of Economics, Wrocław, Poland and Macquarie University, Sydney, Australia*
Marcin Paprzycki, *Systems Research Institute Polish Academy of Sciences, Warsaw Poland and Management Academy, Warsaw, Poland*

Position Papers of the 2019 Federated Conference on Computer Science and Information Systems (FedCSIS)

September 1–4, 2019. Leipzig, Germany

TABLE OF CONTENTS

ARTIFICIAL INTELLIGENCE AND APPLICATIONS

12TH INTERNATIONAL WORKSHOP ON COMPUTATIONAL OPTIMIZATION

COMPUTER SCIENCE & SYSTEMS

12TH WORKSHOP ON COMPUTER ASPECTS OF NUMERICAL ALGORITHMS

6TH INTERNATIONAL CONFERENCE ON CRYPTOGRAPHY AND SECURITY SYSTEMS

7th Workshop on Advances in Programming Languages

4th International Workshop on Language Technologies and Applications

Network Systems and Applications

3rd Workshop on Internet of Things - Enablers, Challenges and Applications

Information Systems and Technology

1st Special Session on Data Science in Health

25th Conference on Knowledge Acquisition and Management

6th Doctoral Symposium on Recent Advances in Information Technology

12th International Workshop on Computational Optimization

MANY real world problems arising in engineering, economics, medicine and other domains can be formulated as optimization tasks. These problems are frequently characterized by non-convex, non-differentiable, discontinuous, noisy or dynamic objective functions and constraints which ask for adequate computational methods.

The aim of this workshop is to stimulate the communication between researchers working on different fields of optimization and practitioners who need reliable and efficient computational optimization methods.

TOPICS

The list of topics includes, but is not limited to:

- combinatorial and continuous global optimization
- unconstrained and constrained optimization
- multiobjective and robust optimization
- optimization in dynamic and/or noisy environments
- optimization on graphs
- large-scale optimization, in parallel and distributed computational environments
- meta-heuristics for optimization, nature-inspired approaches and any other derivative-free methods
- exact/heuristic hybrid methods, involving natural computing techniques and other global and local optimization methods
- numerical and heuristic methods for modeling

The applications of interest are included in the list below, but are not limited to:

- classical operational research problems (knapsack, traveling salesman, etc)
- computational biology and distance geometry
- data mining and knowledge discovery
- human motion simulations; crowd simulations
- industrial applications
- optimization in statistics, econometrics, finance, physics, chemistry, biology, medicine, and engineering
- environment modeling and optimization

BEST PAPER AWARD

The best WCO'19 paper will be awarded during the social dinner of FedCSIS 2019.

The best paper will be selected by WCO'19 co-Chairs by taking into consideration the scores suggested by the reviewers, as well as the quality of the given oral presentation.

EVENT CHAIRS

- **Fidanova, Stefka,** Bulgarian Academy of Sciences, Bulgaria
- **Mucherino, Antonio,** INRIA, France
- **Zaharie, Daniela,** West University of Timisoara, Romania

PROGRAM COMMITTEE

- **Abud, Germano,** Universidade Federal de Uberlândia, Brazil
- **Andrei, Stefan**
- **Bonates, Tibérius,** Universidade Federal do Ceará, Brazil
- **Breaban, Mihaela**
- **Gruber, Aritanan**
- **hadj salem, khadija,** University of Tours - LIFAT Laboratory, France
- **Hosobe, Hiroshi,** Hosei University, Japan
- **Lavor, Carlile,** IMECC-UNICAMP, Brazil
- **Marin, Mircea**
- **Micota, Flavia,** West University of Timisora, Romania
- **Muscalagiu, Ionel,** Politehnica University Timisoara, Romania
- **Pintea, Camelia,** Tehnical University Cluj-Napoca, Romania
- **Stefanov, Stefan,** South-West University Neofit Rilski, Bulgaria
- **Stoean, Catalin,** University of Craiova, Romania
- **Wang, Yifei**
- **Zilinskas, Antanas,** Vilnius University, Lithuania

Position Papers of the Federated Conference on
Computer Science and Information Systems pp. 3–7
DOI: 10.15439/2019F333
ISSN 2300-5963 ACSIS, Vol. 19

Emergency planning and optimizations based on dam break flood risk maps visualized with open source web-GIS tool

Nina Dobrinkova, Stefan Stefanov, Stefan Hadjitodorov
Institute of Information and Communication Technologies – BAS, Bulgaria
Email: ninabox2002@gmail.com, stefans.stefanov303@gmail.com
Center for National Security and Defence Research, Bulgaria
Email: sthadj@cu.bas.bg

Alexander Arakelyan, Alen Amirkhanian, Ara Barseghyan, Susan Mnatsakanian
American University in Armenia
Email: alexander.arakelyan@aua.am, alen@aua.am
Ministry of Emergency Situations in Armenia

George Drakatos, Christos Evangelidis, Vangelis Katsaros, George Boustras
National Observatory of Athens – Greece, Email: g.drakat@noa.gr, cevan@noa.gr
European University Cyprus – Cyprus, Email: ekatsaro@gmail.com, g.boustras@euc.ac.cy

Abstract— **Nowadays technologies are changing every day and with them all services and tools in cases of disaster situations increase. However some sectors such as emergency planning and response are still having difficulties to implement the new technologies. In our paper we will present an idea on how new technologies in flood risk mapping visualization can give more options to the first responders and optimize their time for reaction. The test area is located in Armenia, where exist a special dam constructed for mining purposes. It is built in earthquake vulnerable area and we evaluate the risk of dam break at that location. The final results which are flood risk maps are implemented in specially developed open source web-GIS tool. This tool is applicable for decision making in operational room or any other first responder facilities.**

I. Introduction

THE Alliance for Disaster Risk Reduction project (ALTER project) has been designed in the framework of DG ECHO external line call. These types of projects have as main goal to address cooperation between EU and third party countries. Main idea is best practices transfer from EU to external neighboring country. In ALTER project that selected country is Armenia. The project has as main objective to create public private partnerships to increase resilience in areas of Armenia that face risks from floods originating in earthquakes. Methods, tools, know how and experience from Greece, Bulgaria and Cyprus have been shared with Armenian partners. The partnership of the Armenian government and local stakeholders gave an opportunity to the consortia to work on larger scale at the selected test areas. The project is focused on three pilot areas in Armenia where dams and other activities such as mining processes are presenting the risks to local communities. The areas are: Akhtala and Teghut areas of Lori Marz along the Shamlugh river; the Vorotan Cascade and its associated dams in the Syunik region; and the Voghji river basin of Syunik region. In the paper will be presented information about the Armenian study area. Data that has been collected for dams at this study area. Calculations about potential dam failure and possible flood risk maps. As a final section will be summarized a specifically developed open source web-GIS tool and its main functionalities. The purpose of the web tool is to support a decision making on the field and to optimize resources allocation.

II. Study Area

One of the activities of the project ALTER was to identify the most suitable best practices on risks related to dams in earthquake zones available within and outside the consortia. The study area selected for Armenia was: Kapan and Voghji River Basin. This area is located about 300 km southeast of Yerevan and has a population of about 45,000. It contains some of Armenia's most intensive mining activities and two of Armenia's largest tailing dams – Artsvanik and Geghanush. Additionally, the Geghi Reservoir upstream of Kapan were also included. The villages Kavchut, Andiokavan, Hamletavan, Shgharjik, Syunik and the Kapan Town are located in the immediate floodplain of the Geghi and Voghji Rivers. The village of Verin Giratagh and Nerkin Giratagh are not in the floodplain, however the only road access to these villages is through the floodplain below the Geghi dam. The two tailing dams also pose risk to Kapan's airport which would be needed in case of an emergency and the main highway connecting Armenia and Iran.

The Geghi reservoir is located in Syunik, the southernmost province of Armenia (Figure 1). The reservoir is situated on the Geghi river, the left-bank tributary of the river Voghji. The maximum water level discharge occurs during the spring. Due to the high altitude nature of the area, snowmelt increases gradually as does the level of the river and the reservoir. Snowmelt typically occurs from March to August (Armenian State Hydrometeorological and Monitoring Service).

This work has been supported by the DG ECHO project called: "Alliance for disaster Risk Reduction in Armenia" with acronym: ALTER and Grand Number: 783214.

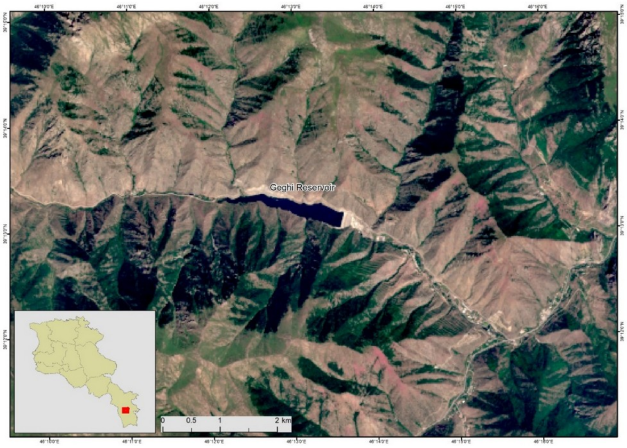


Figure 1. The location of Geghi reservoir. The inset shows its location within Armenia. Background image: Sentinel-2, RGB composite.

The surface of the Geghi reservoir is 50 ha and the elevation above sea level is nearly 1400 m. The height of the dam is 70 m and the length along the crest is 270 m. The total volume of reservoir is 15 million m^3, but the effective volume is about 12 million m^3 [1]. Nearly 4,300 people would be affected by a dam break affecting the reservoir [2]. Geghanoush Tailing Storage Facility (TSF)

Geghanoush TSF is located in the gorge of the Geghanoush River, in the southern part of Kapan (Figure 2). The difference of relative heights between the tailing dam, on one hand, and city buildings and transport infrastructure, on the other hand, is 75 meters. In case the reservoir dam is broken due to an earthquake, the sliding mass could cover industrial and residential buildings, and as a result of barrage, the polluted water could flood central quarters of the city.

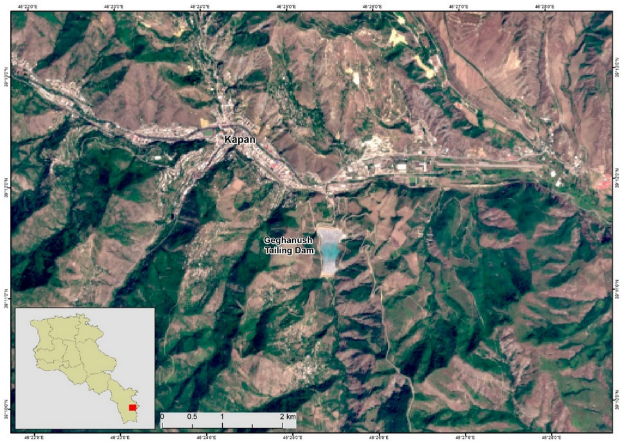


Figure 2. The location of Geghanush Tailing Dam. The inset shows its location within Armenia. Background image: Sentinel-2, RGB composite.

The existing Geghanoush Tailings Repository was designed in early 1960's and had been operated between 1962 and 1983, when the Kajaran Tailings Repository at Artsvanik was commissioned. The Geghanoush tailings repository was re-commissioned in 2006 after the completion of the diversion works and continues to be used today along with an upstream extension currently under construction. The volume of the tailing is 5.4 million m^3 and the dam height is 21.5 m [1].

Tailing and water dams in the appointed pilot area are hazardous hydro-technical structures because of their location in earthquake prone zone. In addition, dam break could occur due to the technical condition of the dams and improper exploitation. Catastrophic flood are possible in this place caused by dam failure. Therefore, the assessment of dam break consequences has a crucial meaning for emergency management and development for measures and action plans for stakeholders and respective authorities in Armenia.

III. Methodology and Data Used

A. Methodology

Flood modeling basics refer to 1D and 2D models, which provide steady and unsteady flows, including the necessity of Manning N values usage.

There are many event types and phenomena that can lead to dam failure:

- Flood event
- Landslide
- Earthquake
- Foundation failure
- Structural failure
- Piping/seepage (internal and underneath the dam)
- Rapid drawdown of pool
- Planned removal
- Terrorism act

Given the different mechanisms that cause dam failures, there can be several possible ways dam may fail for a given driving force/mechanism. In 1985 and in 2002 has been analyzed a list of dam types [3,4] versus possible modes of failure.

The reports from 1985 noticed that of all dam failures – 34% were caused by overtopping, 30% due to foundation defects, 28% from piping and seepage, and 8% from other modes of failure. In the same report of dam failures are included earth/embankment dams, for which 35% have failed due to overtopping, 38% from piping and seepage, 21% from foundation defects, and 6% from other failure modes.

In our work we are doing analysis of a potential dam failure. The prediction of the reservoir outflow hydrograph and the routing of that hydrograph through the downstream valley are evaluated to determine dam failure consequences. There are calculated results about the risk of the population located close to the dam, it is important to accurately predict the breach outflow hydrograph and its timing relative to events in the failure process that could trigger the start of evacuation efforts [5].

B. Hydro-meteorological Observation Data

Hydro-meteorological Observation Data Flood formation and its behavior is highly dependent from hydro-meteorological conditions of the territory. Rainfall intensity and duration, snowmelt, air temperature and other meteorological factors are key drivers in flood development process. Hydro-meteorological monitoring within the territory of Armenia is conducted by Hydromet Service of the Ministry of Emergency Situations of Armenia.

There are 2 operational meteorological stations within Voghji River Basin: Kajaran and Kapan providing information. Their location is presented in table 1.

№	Name of Station	Latitude	Longitude	H, m	Observation Period
1	Kajaran	39° 09' 10"	46° 09' 33"	1843	1975 – present
2	Kapan	39° 12' 15"	46° 27' 44"	705	1936 – present

Table 1. Operational Monitoring Stations within Voghji River Basin

Thermal conditions normally decrease in the Voghji Basin as altitude increases. Multiyear annual average air temperature is in Kajaran is 6.8°C and in Kapan is 12.3°C (Table 2).

Table 2. Annual and monthly average air temperatures in the Voghji River Basin, oC

Meteorological Station	Absolute Altitude (m)	Month												Year
		I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	
Kajaran	1980	-3.4	-3.0	0.5	5.7	10.2	14.2	17.1	16.6	13.3	8.2	3.2	- 1.0	**6.8**
Kapan	704	0.8	2.4	6.3	12.3	16.1	20.4	23.7	23.1	19.0	13.0	7.5	2.9	**12.3**

Rainfall generally increases by altitude in the basin (table 3).

Table 3. Intra-annual distribution of atmospheric precipitation in the Voghji River Basin, mm

Meteorological station	Absolute Altitude (m)	Month												Year
		I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	
Kajaran	1980	44	51	74	84	85	49	23	21	31	52	49	41	**605**
Kapan	704	25	31	59	75	94	66	31	28	41	49	40	25	**565**

The average annual relative humidity is 50-60%, and less than 30% at low altitudes (up to 1000 m). Frost-free days vary by altitude – annually from 260 (at the altitude of 700 m) to 50 days (higher than 3000 m). The annual average relative humidity is 60-80% (over 2600 m), and at lower altitudes - up to 30% (up to 1000 m).

Permanent snow cover starts at altitudes of 1200 m and it lasts for 35-165 days. The snow depth is 15-180 cm. It lasts 1-1.5 months at altitudes of up to 1500 m, and 6.5-7 months at altitudes of 3000 m and higher. The depth of snow cover is 15-20 cm at altitudes of 1300-1500 m and 120-180 cm at altitudes of 3000 m and higher (from place to place a 300 cm thick snow cover is formed, due to winds occurring in concavities).

Evaporation drops to 482-220 mm as altitude increases in the Voghji River Basin. The highest value of evaporation, 500-480 mm,. is observed at altitudes up to 800 m.

There are 3 operational hydrological monitoring posts within Voghji River Basin: Voghji-Kajaran, Voghji-Kapan and Geghi-Kavchut. Data of closed monitoring posts of Geghi-Geghi and Geghanoush-Geghanoush were analyzed as well due to their importance for the Geghi reservoir and Geghanoush tailings dam break modeling (tables 4 and 5).

Table 4. Hydrological Monitoring Posts within Voghji River Basin

№	Water Object Name	Name of station	Coordinates	
			Latitude	Longitude
1	Voghji River	Kajaran	39° 08' 59"	46 09' 16"
2	Voghji River	Kapan	39° 12' 18"	46 24' 43"
3	Geghi River	Kavchut	39° 12' 23"	46 14' 50"
4	Geghi River	Geghi	39° 13' 21"	46 9' 36"
5	Geghanoush River	Geghanoush	39° 10' 35"	46 25' 24"

Table 5. Flow Characteristics in the Hydrological Monitoring Posts within Geghi River Basin

River-Post	Discharge, m³/s													
	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	Annual Average	Maximum
Geghi-Geghi	1.5	1.5	1.9	5.5	12.7	12.9	6.9	3.3	2.3	2.0	1.8	1.6	4.5	37.7
Geghi-Kavchut	1.4	1.5	2.3	5.9	12.5	12.0	6.3	3.0	2.1	1.9	1.7	1.5	4.4	87.5
Geghanoush-Geghanoush	0.2	0.2	0.7	1.9	1.8	0.9	0.4	0.3	0.3	0.3	0.3	0.2	0.6	21.3
Voghji-Kajaran	0.5	0.5	0.8	3.0	7.7	11.5	7.2	2.5	1.0	0.7	0.6	0.5	3.0	43.9
Voghji-Kapan	2.4	2.6	4.6	14.5	28.7	28.5	15.6	6.4	3.9	3.6	3.2	2.7	9.7	270.0

A. Elevation Data

Elevation data has a crucial meaning in each flood modeling process. There are various free digital elevation models (DEMs) available online (SRTM, ASTER, ALOS), the spatial resolution of which is ~30 m. This resolution is not enough for detailed flood mapping in mountainous areas.

Georisk CJSC provided linear shapefile of elevation isolines of 1:10,000 scale. From this shapefile, 5 m resolution DEM of studied area was calculated using Topo to Raster interpolation tool of ArcGIS Spatial Analyst toolbox (Figure 3):

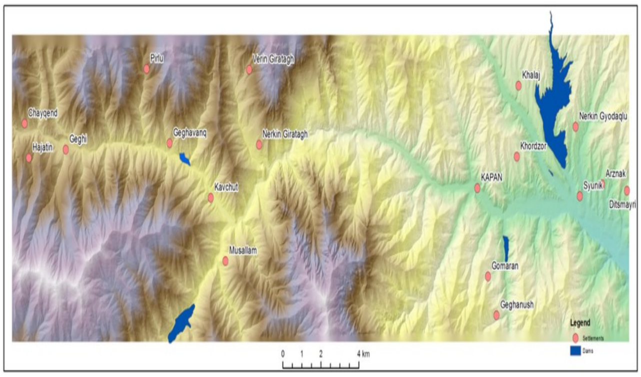


Figure 3. 5m DEM of Studied Area

Geomorphometric parameters (slope, aspect and shaded relief) were derived from DEM and by using ArcHydro Tools, raster layers were calculated from the DEM (Filled DEM (hydrologically-corrected); Flow Direction; Flow Accumulation; Streams (defined and segmented); Catchments GRID).

Catchment polygon has been created based on the layers created and drainage line vector layers were obtained. A detailed land-cover and land-use maps have been created using the European Space Agency (ESA) and the European Commission services [6]. For open water (including Geghi reservoir) and tailing ponds Sentinel-1 (SAR) data was used. After obtaining the results of open water and tailing ponds then they were superimposed on the other classes. The final map is shown in the Figure 4.

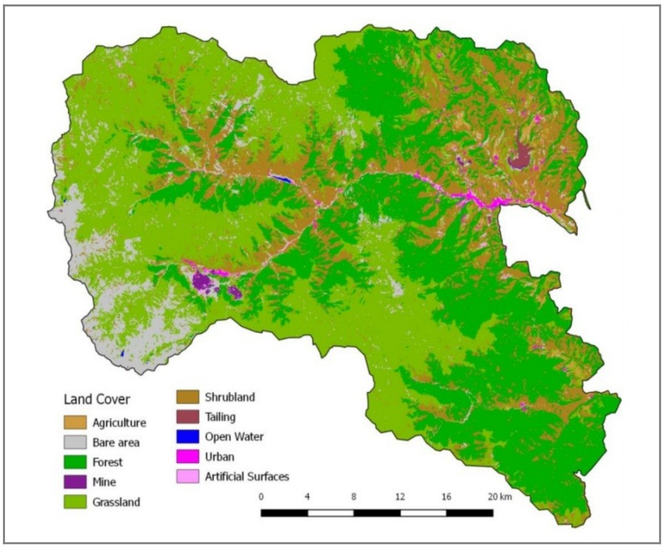


Figure 4. Land-cover and land-use map of Voghji river basin

IV. DESKTOP APPLICATION FOR FLOOD HAZARD AND RISK IN ARMENIA

Based on the flood maps presenting different scenarios of water spill from the dam break a specially developed open source desktop tool has been created. Its architecture is based on model designed within the ALTER consortia with no use of commercial software. The application is mainly based on open source GIS software; server part for the dynamic events, JavaScript and its libraries and frameworks. The tools implemented are open source software solutions such as: Geoserver, Qgis, Web App Builder, Boundless WEBSDK, OpenLayers.

Geoserver allows the user to display spatial information to the world;\newline

QGIS is a professional GIS (Geographic Information System) cross-platform application that is Free and Open Source Software (FOSS);

Web App Builder is a plugin for QGIS that allows easy creation of web applications;

Boundless WEBSDK which provides tools for easy-to-build JavaScript-based web mapping applications;\newline

OpenLayers is an open-source JavaScript library for displaying map data in web browsers.

It includes different features and tools that may lead to faster response and easier way of taking decisions in flood event cases.

The application has the function to visualize the most vulnerable buildings (Figure 5). It includes different scenarios that can be analyzed in operational room and by its tools can support better management of the current and future situation in cases of flood events. It is focused on visualization of high waves coming after dam break in cases of failure.

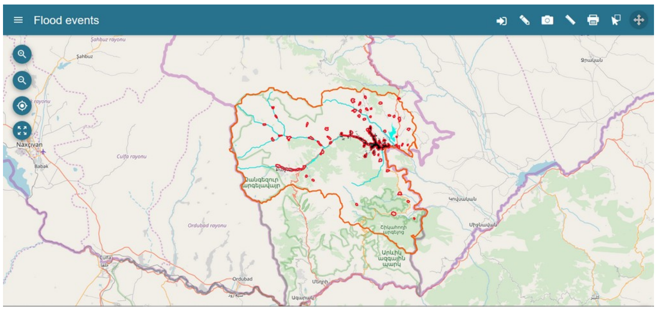


Figure 5. Application main screen

The application has the ability of switching the predefined layers and the base map layers. The predefined layers have a very rich data by turning them off or on. Users can easily make analysis of the risks in cases of flood events. It includes different scenarios of the water spread in support of better decision making and faster resource allocation. Layers can be downloaded as geojson files. Geolocation of team members on the field is available for the users.

The base map layers are including Street map, Satellite map, Shaded relief map and NatGeo map which can be used in operational room analysis.

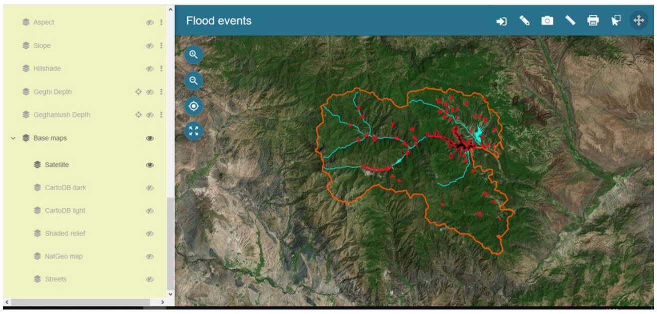


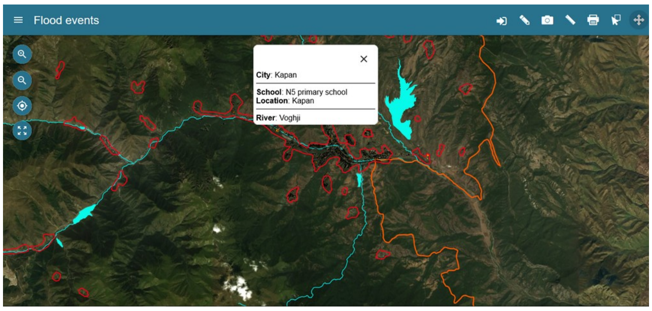
Figure 6. Layers list

Draw feature tool can mark the zone of interest (Figure 6) by polygon or line which will be visualized and be seen in the operational room in real time.

The Popup feature visualizes information about the vulnerable buildings such as: schools, kindergartens and others (Figure 7).

The export feature can save maps with new data as picture format files. This feature can be used in future data analysis.

Measure and distance options can be used to measure the distances and also can measure the size of the focused area.



City: Kapan

School: N5 primary school
Location: Kapan

River: Voghji

Figure 7. Application Popup

The application provides connection to the current weather forecast via openweather with detailed information about the current or future weather conditions. It is connected to EFAS emergency management service which provide extra satellite data about current conditions.

V. Conclusion

GIS raster layers of flood inundation zones and depths, as well as tables of simulated dam break characteristics for Geghi Reservoir and Geghanoush TSF Dams were developed for three failure scenarios: full failure, half failure and 10% failure. Maximum depth of the flooding, maximum absolute altitude, flooding time, maximum flow discharge and velocity in given cross-section and other parameters were calculated. Based on the analysis and discussions of these results open source web-GIS visualization tool has been developed and implemented for testing the areas of the research work.

References

[1] Georisk CJSC (2017) Assessment of the Multi-Component Risk Determined by the Maximum Seismic Impact on the Kapan City (Multi-Hazard City Scenario). Project # ARM 10-0000005849, Final Report.

[2] Gevorgyan A., Minasyan R., Khondkaryan V., Antonyan A. (2014) The Prediction of Possible Flooding of the Territory as a Result of the Accident of the Geghi Reservoir Dam.

[3] Costa, John E., 1985. Floods from Dam Failures. United States Department of the Interior, Geological Survey, Open-File Report 85-560, Denver, CO.

[4] Atallah, Tony A., 2002. A Review on Dams and Breach Parameters Estimation. Master of Science in Hydrosystems Engineering, Virginia Polytechnic Institute & State University, Blacksburg, VA, January 2002.

[5] Wahl, Tony L., "Dam Breach Modeling – an Overview of Analysis Methods". Joint Federal Interagency Conference on Sedimentation and Hydrologic Modeling. June 27 - July 1, 2010, Las Vegas, NV.

[6] Schlaffer S., Harutyunyan A. (2018) Working Paper: LCLU Voghji River Basin AUA Acopian Center for the Environment, AUA GIS and Remote Sensing Lab.

Position Papers of the Federated Conference on Computer Science and Information Systems pp. 9–17
DOI: 10.15439/2019F356
ISSN 2300-5963 ACSIS, Vol. 19

A Heterogeneous Parallel Processing System Based on Virtual Multi-Bus Connection Network

Piotr Hajder, Łukasz Rauch
AGH University of Science and Technology, Krakow, Poland
Email: {phajder, lrauch}@agh.edu.pl

Mariusz Nycz
Rzeszow University of Technology, Rzeszow, Poland
Email: mnycz@prz.edu.pl

Mirosław Hajder
University of Information Technology and Management
Rzeszow, Poland
Email: miroslaw.hajder@gmail.com

Abstract—**This work presents organization, architecture and synthesis as well as analysis of the reconfigurable heterogeneous parallel processing system. Reconfiguration takes place on two levels of the connection network: physical and logical. For its implementation, passive multi-channel optical networks were used. Due to its dynamic nature, the system is designed to handle computational and communication load of an explosive nature and is addressed in the first place to the production sphere of economy. The dynamically combined connection network not only prevents traffic bursts, but also based on the physical and logical circuit commutation gives the possibility of adapting to the existing traffic pattern. Although the described solution is addressed to the optical transmission environment, its effective functioning in the Ethernet networks with circuit switching and partly in wireless networks has been confirmed empirically. The theoretical foundations were verified in the design and construction of a reconfigurable super-microcomputer and the intelligent system detection of attacks addressed to industrial Internet.**

I. Introduction

THE progressive globalization of information systems and dynamic development of data transmission techniques have resulted in significant changes in the use of information systems and parallel computing. While earlier systems and parallel processing were the domain of scientific centers, they are now also widely used in industry and management. It is difficult to talk about any comparison of computer used in science and industry; however, they share a common feature: the parallelization of the computational process, which is implemented on many devices.

The performance of a parallel computer system is primarily determined by the performance of its components: computational elements and connections between them. Theoretically, the communication subsystem has less impact on the performance, but in practice it determines the size of entire system and thus its final efficiency. Computational efficiency, also referred as its performance, is a quantitative feature of the speed at which specific operations are performed. In turn, the most important parameter of the communication subsystem is its bandwidth defining the number of information units sent over time unit.

In the book [1] considered to be classical, the mathematical and methodological basis for estimating the above parameters has been comprehensively presented. Similar considerations for computer systems, networks and cloud computing can be found in [2]-[5]. This subject can be found in the latest publications in conference materials and magazines. They focus on determining power in heterogeneous systems, including IoT [6]-[8]. Solutions of this type are used in computations in the area of basic research, nuclear physics, medicine, and so on.

Assigning to all computer systems a statement known from the literature, that the only criterion for their evaluation is computational efficiency they offer, is inappropriate [9]. In supercomputers and clouds, the maximum efficiency is indeed an important factor of the entire solution [10], [11]. In other classes of systems, other properties may be desirable. For example, in management systems, information gathering, industrial IT, control systems, power and bandwidth should correspond to the actual demand and other characteristics or parameters are particularly desirable. The following statements can be distinguished:

a. In industrial IT crucial system properties change over time, including efficiency, bandwidth, availability, latency, etc.;
b. The effectiveness of inter-communication depends on the relationship between topology and supported traffic pattern, which is not constant and can change dynamically over time;
c. The size of computations made, and the volume of data transferred per time unit vary and may temporarily have a bursty character that can cause system's malfunctions;
d. In the system exist temporarily underloaded units that can participate in computations within any part of the system and tasks being solved.

From an economic point of view, the solution to the problem of insufficient resources cannot be handled by their redundancy as it generates significant costs both at the construction and operation stage. Most likely, these resources would be never used in satisfactory manner [9], [12].

In this work, the problems of insufficiency and redundancy of resources are solved by self-adjusting communication network based on multi-channeling, grouping and hierarchization. It is assumed that parameters and characteristics of the communication network are most important properties of entire computer system.

In the classification of the topology optimization methods presented in [13] among reconfigurable networks, there are demand-aware networks in which the adaptation to the load can be static or dynamic. In the first case, the topology is immutable while in the second one is reconfigurable. Reconfigurable networks are also called self-adjusting. Additionally, it can be noted that although reconfigurable architectures represent an interesting development paradigm, there are currently no analytical tools to study their potential [13].

The concept of distributed passive commutation developed in this work was proposed in [14] and the organization of experiments conducted is described and justified in [15]. The main goal is to present research on creating a budget reconfigurable heterogeneous system intended for the industry.

The paper starts with section 2, where main idea of the virtual multi-bus connection system and fundamental assumptions are presented. This section ends with classification of the communication efficiency improvement methods and the design task definition.

In section 3, a description of the design tasks is described and the methodology for designing a parallel processing system based on the virtual multi-bus connection system is provided. Three independent optimization criteria were considered: reliability, efficiency (computational and communication) and latency. For each, an appropriate algorithm were developed.

The paper ends with description of simulation tests performed, conclusions and further research goals.

II. Preliminary arrangements

A. Bus Systems and Their Representation

Let's assume that a parallel computer system with bus communication is a combination of two types of equal objects: computational nodes N and buses B. A node can be incidental with any number of buses. A network composed of n nodes and m buses is usually denoted as $[n, m]$ and describes an incident matrix $I = \{b_{ij}\}$ of size $n \times m$. The element $b_{ij} \in I$ is equal to 1 if the node N_i, $i = 1, \dots, n$ is incidental with bus $B_j, j = 1, \dots, m$, otherwise $b_{ij} = 0$. By the definition of incident matrix, bus networks do not allow loops for both buses and nodes. Therefore, if there are multiple connections between any two elements in the system (i.e. the selected node will be integrated with the selected bus by means of several connections), the traditional method of bus denotation cannot be used.

If the number of buses incident with i-th node is designated as s_i^w, $i = 1, \dots, n$ (node degree) and the number of nodes incident with j-th bus as s_j^m (bus degree), then for any bus network $[n, m]$ there is relationship between the total degree of nodes and buses:

$$\sum_{i=1}^{n} s_i^w = \sum_{j=0}^{m} s_j^m = s \,. \tag{1}$$

Expression (1) is the basis of network synthesis method developed by authors with single connections presented among others in [16], [17]. In contrast to the network with direct connections, for the bus network $[n, m]$ with incidence matrix M, there is always a network with a transposed incidence matrix M^T.

Bus networks are structurally equivalent to hypergraphs. They are used to analyze bipartite graphs [18]. The number of nodes in X_0 and X_1 parts of bipartite graph is equal to n and m, respectively. Its edges are local connections $l_{p,q}$ where p, q – the number of computational node and the bus respectively, whose purpose is to connect the computing node with the bus. This is, in fact, a description of the PBL neighborhood graph (**P**rocessor-**B**us-**L**ink).

The PBL graph $G = (V, B, L)$ containing $|V_G| = n$ nodes, $|M_G| = m$ buses and L_G link set is the bipartite graph G_{PBL} which can be described by following pair: $\left(V_{G_{PBL}}, B_{G_{PBL}}\right)$ and $V_{G_{PBL}} = VV_{G_{PBL}} \cup VB_{G_{PBL}}$, where $VV_{G_{PBL}} = V_G$ and $VB_{G_{PBL}} = B_G$. Connections in graph G between buses and nodes are represented by $B_{G_{PBL}}$. The nodes $v_{G_{PBL},i}$ and $b_{G_{PBL},j}$ are connected by the edge $l_{G_{PBL},k}$ if and only if in the source bus system, the bus b_i is connected to the node v_j with link l_k. Such a representation of the bus system is shown in Fig. 1.

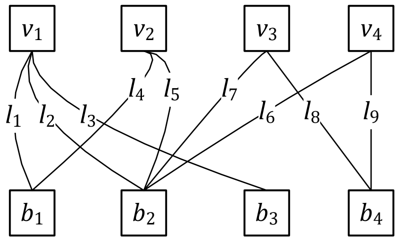


Fig. 1. The general form of multi-bus system

To describe graph in simulation programs a neighborhood matrix with a block structure was used:

$$A = \begin{bmatrix} 0_{n,n} & W \\ W^T & 0_{m,m} \end{bmatrix}, \tag{2}$$

where A – matrix with size $n \times m$, $0_{n,n}$, $0_{m,m}$ – zero matrices with size $n \times n$ and $m \times m$ respectively. When storing in computer's memory, these elements are omitted and all information about the graph is contained in the sub-matrix W, sometimes called bi-neighborhood matrix. For a bipartite graph G with parts $V = \{v_1, \dots, v_n\}$ and $B = \{b_1, \dots, b_m\}$ the bi-neighborhood matrix W is a binary matrix of size $n \times m$ where $w_{i,j} = 1$ if and only if $\left(v_i, b_j\right) \in L$ [19].

The designed parallel processing system is heterogeneous, consists of server units with relatively large computing power, and dynamically connected low-efficient Raspberry, Arduino or similar units. Therefore, for the mathematical description of the system tripartite graph was used, whose components are servers S, support servers K and buses B. It was presented on Fig. 2.

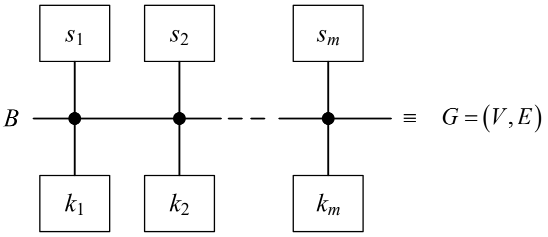


Fig. 2. Multi-bus system presented as tripartite graph

The denotation of tripartite graph in the matrix form was proposed in [18], [20]. Their representation is reduced to a diagonal graph, then presented as a subgraph of an upper graph which is, in fact, an algebraic graph.

As an alternative method for describing the bus network topology, graph algebra was developed for this purpose [20]. Let sets $S=\{s_1,s_2,\dots,s_r\}$, $B=\{B_1,B_2,\dots,B_m\}$, $K=\{k_1,k_2,\dots,k_n\}$ describe sets of primary servers, referred to as servers, buses and support servers, respectively. Informally, connections between elements can be described using following rules:

1. The recipient prefers the selected service provider. Preferences are not permanent and can be changed without restrictions during the work of Multi-bus Reconfigurable Computer System;
2. Connection between service recipients and service providers is performed by means of logical bus channel.

The formalization of this process takes place in the following way. Let's define a threefold relation $P \subseteq K \times S \times B$ on sets K,S,B: $(k,s,b) \in P \Leftrightarrow k$ prefers the service provider and these preferences are carried out using bus b $(k \in K, s \in S, b \in B)$. Relation P induces two relations of equivalence R and R^- on sets K and B respectively:

$$k_i R k_j \Leftrightarrow (k_i,s',B'),(k_j,s'',B'') \in P \text{ i } s'=s'', \tag{3}$$

$$\begin{gathered} b_i \overline{R} b_j \Leftrightarrow (k_i,s',b_i),(k_i,s',b_i),(k_s,s'',b_j) \in P \\ s'=s'' \end{gathered} \tag{4}$$

Informally, the relations R and R^- mean that the supporting servers k_i and k_j prefer the same server and joining it is carried out using the b_i and b_j buses. Connecting the selected server to the supporting server is performed by means of one and the same bus, i.e. $k_i\, Rk_j \Rightarrow b_t = b_l$, $i,j=1,\dots n$, $t,l=1,\dots,m$ and $k_i \neq k_l \Rightarrow s_i \neq s_j$. The model in the form of finite algebra of undirected connected graph can be described as:

$$\begin{aligned} &\left(T^0_{s_1} * T^0_{B_1} \cup T^0_{B_1} * T^0_{B_1}\right) *_{f_1} T^0_{B_2} \cup \dots \\ &\cup \left(T^0_{s_{m-1}} * T^0_{M_{m-1}} \cup T^0_{M_{m-1}} * T^0_{k_{n-1}}\right) * \\ &*_{f_{n-1}} T^0_{M_m} \cup \left(T^0_{s_m} * T^0_{B_m} \cup T^0_{B_m} * T^0_{k_m}\right) = \\ &= \left(T^0_{s_1} * T^0_{B_1} \cup T^0_{B_1} * T^0_{k_1}\right) \cup \dots \cup \\ &\cup \left(T^0_{s_m} * T^0_{B_m} \cup T^0_{B_m} * T^0_{k_m}\right) *_{f_1} T^0_{B_1} *_{f_2} \dots *_{f_{m-1}} T^0_{B_m} = \\ &\left(T^0_{s_1 B_1} \cup T_{B_1 k_1} \cup T_{s_2 B_2} \cup T_{B_2 k_2} \cup \dots \cup T_{s_m B_m} \cup T_{B_m k_m}\right) * \\ &* f_1 T^0_{B_1} * f_2 \dots * f_{m-1} T^0_{B_m}. \end{aligned} \tag{5}$$

Due to the alternation of connection operations and mutually unambiguous connection, it can be noticed that service providers and buses are equal and can be interchanged. The graphical representation of the organization described by expression (5) is shown in the form of tripartite graph in Fig. 2. The element ensuring its consistency is bus set B.

B. *Organization of the Multi-bus System*

The organization of a parallel reconfigurable computing system with multi-bus connections is shown in Fig. 3.

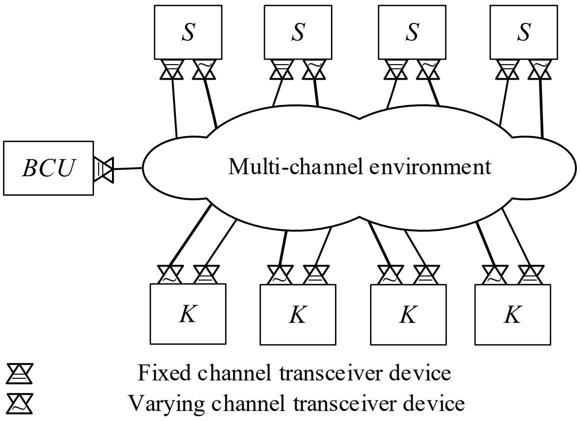

S
S
S
S
BCU
Multi-channel environment
K
K
K
K
Fixed channel transceiver device
Varying channel transceiver device

Fig. 3. The basic organization of parallel reconfigurable computing system: BCU – bus control unit.

Each of the computing nodes has been equipped with at least one fixed and one adjustable single-channel communication interface. If it is possible, the parameters of each fixed channels are unique in the whole organization, not limiting to setting up networks of any connection architecture. Deviation from the above rule is the management channel which all computational nodes are connected to. This channel always uses broadcast transmission and is intended only for sending information on the configuration of adjustable transceiver devices.

In the design of the connection system following assumptions were made:

1. The purpose of using multi-channel connections is to minimize the complexity of the design, construction and operation of computer systems using a set of processing elements. Instead of direct connections (which ale complex in design), broadcasting logical channels are used defining the architecture which is

relatively simple. In addition, multi-channeling should ensure maximum efficiency of using the bandwidth available in the physical transmission channel. Communication channels can work not only in the broadcast mode, but also two-point connections are possible. The only channel connected to all nodes and working only in broadcast mode is management channel. Depending on the acceptable construction and operating costs, it can be made in physical or logical form;

2. The bus channel can be built in the form of an active electronic or passive optical system. The use of all possibilities offered by the proposed architecture is possible in the second case. The proposed architecture, to a limited extent, can be made based on classic network switches equipped with a circuit switching. If it is acceptable to build a network with group broadcasting, the functional scope of the prototype increases. In the electrical implementation of the architecture, the management channel is redundant, because its role is fulfilled by the control systems of a switch. Part of the system's functions with proposed architecture can also be reconstructed based on wireless networks;
3. Each server is equipped with one fixed channel transceiver and several adjustable devices working with any channel available in the system. The greater the number of interfaces, the system capabilities are wider. The disadvantage of the tunable devices is the relatively long retuning time, as well as the higher purchase cost (theoretically). In turn, the use of fixed interfaces limits the possibilities of reconfiguration;
4. Logical channels are build based on the resources of physical communication channels. Physical channels use bus or star topology. The transmission in physical channel has a passive directed character. A broadcast transmission mode is used in the logical channel;
5. Reconfiguration consists in tuning the transceiver devices to work with another logical channel. This process is initiated by BCU and implemented by means of broadcast control bus;
6. BCU managing node, based on information flow and node load, designs in real time a new architecture of links and sends information about it to every computational node. Since each of the elements must be permanently connected to the management node, the use of fixed transceiver device is preferable;
7. Each node performs tuning operation on its transceiver interfaces, thus creating a new connection network. The condition for connecting the processing node to any other node or group of them is to work with the same logical channel available within the same physical channel;
8. In the presented architecture it is assumed that channel splitters (couplers) are passive devices. Further functionalities of the system expanding the scope of its reconfiguration can be obtained by using active devices. Their functionality has been proposed by the authors, among others in [14], [15]. In particular, it is interesting to divide logical buses into fragments, as has been used for physical ones.

From the point of view of computer systems theory, reconfigurable bus systems should be classified into a group of reconfigurable multi-machine systems with tight or loose (more often) connections. Systems with tight connections will be called those in which transceiver elements will use the same type of logical channel. On the other hand, in the systems with loose connections transceiver components of the computational node are completely independent of each other. Theoretically, when designing a system both tight and loose connections can be used. In the analyzed solutions, in order to minimize costs, mainly organizations with a single connection of a computing element with a logical bus are used. In addition, transceiver devices should work with one type of logical channel. Otherwise, the bus input and output channels would be separated, and computational elements would be used for communication between nodes. Therefore, in the analyzed system it is recommended to use tight connections.

If the computational node is connected to the logical bus via several transceiver elements, there is a theoretical possibility of splitting the transmitter and signal receiver, and thus creating systems with loose connections. The above hypothesis, however, rises some doubts. First, the logical bus must use the same channel along its entire length. Secondly, connecting the processing node to the logical bus via several transceiver devices is only appropriate if the system is more resistant to failures. Thus, from the point of view of system definition with loose and tight connections, the analyzed systems are characterized by tight connections. Single, multiple and partial connections of computational element (service provider or recipient) with a logical bus is schematically shown in Fig. 4.

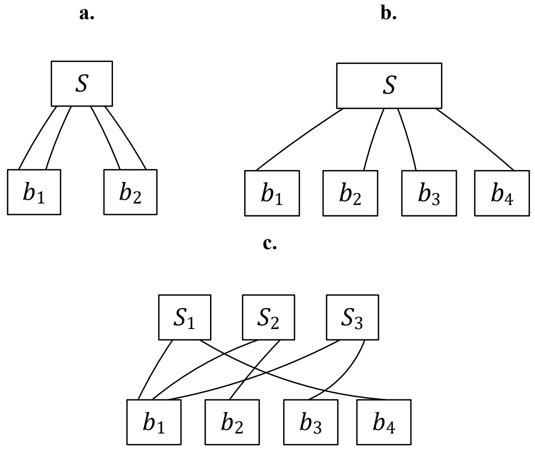


Fig. 4. Node connections to the virtual bus: a. Multiple (double) complete; b. Single complete; c. Partial

Let's consider the condition of the computer system's readiness with equivalent nodes. Let K_{in} specify the total number of processing nodes used in the system and K_{in}^{min}

– the minimum number of nodes necessary to implement all its functionalities and k_{in} – number of correctly working nodes in the system. Then, the condition of readiness has the form: $K_{in} \geq k_{in} \geq k_{in}^{min}$. In addition, it is necessary to operate a communication subsystem that ensures interoperability of at least k_{in}^{min} processing nodes. For systems with a server and supporting server, the following designations are used: K_k – the total number of supporting server; K_s – total number of servers, k_s^{min} – the minimum number of servers necessary to implement the systems functionality; k_k^{min} – the minimum number of supporting servers necessary to implement the systems functionality; k_s – the number of correctly working supporting servers; k_k – the number of correctly working supporting servers. The readiness condition can be written as: $K_s \geq k_s \geq k_s^{min}$ and $K_k \geq k_k > k_k^{min}$. In addition, the correctness of the communication subsystem ensuring interoperability of no less than k_s^{min} service provides and k_k^{min} service recipients should be guaranteed.

In this research, servers were usually desktop units with extensive configuration. As the supporting server, the most commonly used are self-made network traffic analyzers.

C. Classification of the multi-bus systems

In traditional bus networks, the communication network is single-channel and each of the system's element is connected to it only once. In the case of multi-channel buses, user is usually connected only to a specific channel. An exception to this rule is switchable optical networks in WDM technology, whose implementation costs exclude the possibility of their use in non-telecommunication applications.

Reconfigurable multi-bus parallel systems have several elements to effectively adapt to the demand for computational efficiency and bandwidth. As the first one should be mentioned the possibility to equip both types of computing nodes with a variable number of transceiver elements. This allows one or more times to connect any bus or a set of buses. Secondly, the use of transceiver devices tuning to a specific channel provides dynamic changes in connections and grouping of supporting servers around primary servers. The groups being build (node clusters) have their own communication environment and can be isolated from external interference. For example, a separate group can be created by users using services insensitive to latency, another generating low traffic, yet another characterized by bursty traffic. Thirdly, the buses are also grouped (clustered). In this way, not only the computational efficiency, but also bandwidth of communication channels can scale. The most useful is mixed grouping, where computational nodes and communication channels are simultaneously grouped. Fourthly, the multiplicity of node interfaces allows the use of folded buses (single, double or triple) ensuring better usage of the bus. Another method consists in the hierarchization of logical channels where channels are combined into groups that support different sets of servers. It is also possible to divide overloaded buses into smaller parts.

Thanks to the diversity of the proposed methods, the architecture of connections can be adapted to the traffic patterns and communication requirements. Because in optical systems the change in the wavelength used by transceiver element takes milliseconds, the adaptation of the connection architecture to current requirements can be dynamic and carried out in real time.

The classification of bus interconnection architecture is shown in Fig. 5.

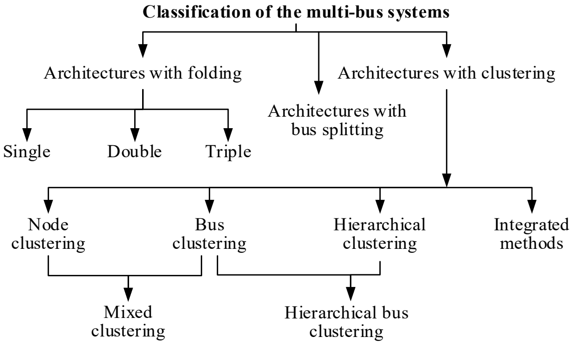

Classification of the multi-bus systems
Architectures with folding
Architectures with clustering
Architectures with bus splitting
Single
Double
Triple
Node clustering
Bus clustering
Hierarchical clustering
Integrated methods
Mixed clustering
Hierarchical bus clustering

Fig. 5.Classification of multi-bus architectures

D. Definition of the project task

For a multi-bus communication system, a connection architecture should be designed to ensure **a.** Maximization of reliability; **b.** maximization of computational efficiency; **c.** minimization of latency while ensuring the minimum (maximum) level of other system characteristics.

Many parameters affect the design process to a greater or lesser extent. The first of them is the generalized construction cost [18], [21]. It is assumed that the generalized costs of constructing the system will be proportional to the number of informational inputs. In a given case, the parameter is the summary logical degree S_Σ^{LF} of all physical (processing) nodes connected to the logical buses used. From a technical point of view S_Σ^{LF} is the total number of transceiver devices used in the nodes of service providers and system users. If the analyzed architecture is based on expensive tunable devices, the value S_Σ^{LF} determines costs of the system. If an optimization design procedure is performed, the generalized costs will be one of the optimization criteria.

The second important parameter is the communication channel's load. In a special case, the system may use two-point communication channels with one server and one supporting server. This case, however, should be regarded as unique. In fact, a set of units of both types will be added to the logical bus, whose efficiency will determine the bus load. The parameter of a system using logical buses is branching factor W_{Ri}. The branching factor of the i-th bus is defined as a sum of transceiver devices in which the processing nodes connected to the common logical bus are equipped. From the point of view of load minimization, it is advantageous to use only single connections (see Fig. 4). In this case, the branching factor W_{Ri} of the i-th logical bus depends only on the number of processing nodes connected to it.

If all the processing nodes load the transmission channel identically, the W_{Ri} factor determines the level of logical bus occupancy. The value W_{Ri} manifests a key influence on the bus channel efficiency, i.e. it determines the size of coupling-splitting physical communication channels. In the presented organization, the processing elements are connected with each other by means of a set of logical buses functioning in one or a set of physical buses.

Connection of the computational element to a specific logical bus is done by setting the parameters of transceiver device. Therefore, from the point of view of performance and bandwidth analysis, parameters related with their reconfiguration time are important. Further, the symbol t_n indicates the time of setting the transceiver element, i.e. the required time to adapt to the indicated logical channel after processing node is started. In turn, t_p will mean the transceiver's device tuning time, i.e. the time required to adapt to specific logical channel. It can be assumed that for the analyzed system with tight connections, the setting and tuning times are identical, i.e. $t_n = t_p$. In a system using tunable transceiver devices, a situation may arise when the currently tuned device should stop the logical channel change and adapt to another channel. As t_s the time necessary to stop the tuning process and start to work with another channel will be marked. It is assumed that $t_s \leq t_p$. If $t_s = 0$, the stopping of tuning process is immediate.

Another important parameter of the computational system, especially for industrial application, is its reliability. The research used the reliability model developed by the authors assuming the equivalence of processing nodes and the interrelationship between physical and logical components of the connection network [15], [18], [22].

III. Design process of multi-bus architecture

A. The Idea of Methodology

Numerous algorithms for designing computational system with multi-channel bus connections are known from the literature. Most of them focus on ensuring a certain level of performance. From the point of view of connection system, these algorithms are aimed at creating an architecture with a specific level of performance or latency occurring between computational elements considering the condition of non-transferability of given costs.

In contrast, the proposed methodology additionally takes into account the level of reliability of computational system. First, it allows to create computing systems with connections characterized by maximum reliability and minimum acceptable communication efficiency and maximum communication delay. Secondly, it ensures the load balancing in bus channels.

Considering the requirements for ensuring minimum acceptable value of bandwidth and reliability as well as the maximum acceptable construction costs and latency, three design tasks can be defined.

B. Task 1: Maximization of reliability

Determine the minimum number K_B of communication channels and the method of connecting computing nodes to them, which ensure the maximum level of reliability with a limited total cost of technical devices, minimum acceptable efficiency D_C and transmission time *t*.

Task 1 can be defined as construction of a maximum reliable computing system, which can be written as an optimization task:

$$R(K_B) \to \max \qquad (6)$$

for the following restrictions:

$$t(K_B) \leq t_{max}\ ,\ D_C(K_B) \geq D_C^{min}\ ,\ U_\Sigma(K_B) \leq U_{max}$$

where: t_{max} – the maximum acceptable value of latency between any pair of computational nodes; D_C^{min} – minimal acceptable computational efficiency of the system; U_{max} – maximum acceptable cost of technical devices.

The task described in (6) should be solved using procedure consisting of three basic steps. To achieve the goal, those steps should be described in the following form:

$$R(K_B^{min}, K_B) \to \max$$

for the following restrictions:

$$1 \leq K_B \leq K_B^{min}\ ,\ t\left(K_B^{min}, K_B\right) \leq t_{max}\ ,$$

$$D_C\left(K_B^{min}, K_B\right) \geq D_C^{min}\ ,\ U_\Sigma\left(K_B^{min}, K_B\right) \leq U_{max}$$

where: K_B^{min} – minimum number of fully operational buses, which ensures the maximum acceptable latency t_{min} and efficiency D_C^{min}. Next, each step will be described.

Step 1. Create the basic system configuration composed of K_B^{min} buses. The sequence of actions performed within a given step is shown in Algorithm 1.

Algorithm 1. The first step of designing reliable connection system

```
input: $D_C^{min}$, $K_B^{max}$, $t_{max}$, $K_{in}$
for $l \leftarrow 1\ to\ K_B^{max}$
    if $(D_C(l) \geq D_C^{min} \wedge t(l) \leq t_{max})$
        if $(l \leq K_{in})$
            return $partialConnections$
        else
            return $completeConnections$
        end if
end for
return $\emptyset$
```

Afterwards, configurations are created containing $l = 1,2,\dots,K_B^{max}$ channels, where K_B^{max} – maximum acceptable number of channels based on technical and economic criteria. Then, using the methods proposed in [13], [15], their performance and latency are evaluated. If $K_B^{min} > 1$. Then, it is assumed that the communication environment is functioning in the load sharing mode between all K_B^{min} channels.

First, complete connections are analyzed in which the number of channels $K_{in} > K_B^{min}$. If any configuration with complete connections that meet all the restrictions cannot be found, architectures with partial connections are created and examined $\left(K_{in} < K_B^{min}\right)$. The searching process ends, when the first configuration that meets the restrictions is found (i.e. with the minimal number of channels K_B^{min}).

Step 2. Improve the reliability of the connection system by introducing additional transceiver elements and verify restrictions on the total cost of the system. The sequence of actions performed at given step is shown in Algorithm 2.

Step 3. Evaluate computational efficiency and latency as well as the reliability of the synthesized system. If these requirements are not met, the design process is repeated using modified inter-node connections classified in Fig. 5.

Algorithm 2. The second step of designing reliable connection system

```
input: $K_{in}$, $U_{max}$
for $K_B \leftarrow K_{in}$ to $K_B^{min}$
  if $(U_\Sigma(K_B) \leq U_{max})$
    return 1
  end if
end for
return 0
```

C. Task 2: Maximizing system efficiency

Determine the minimum number of K_B channels and the method of connecting computing nodes to them, ensuring maximum computational efficiency D_C.

Task 2 can be defined as the task of building computing system with maximum performance. The relevant optimization task has the following form:

$$D_C\left(K_B^{min}, K_B\right) \rightarrow \max$$

for the following restrictions:

$$1 \leq K_B \leq K_B^{min},\ t\left(K_B^{min}, K_B\right) \leq t_{max},$$

$$R\left(K_B^{min}, K_B\right) \geq R_{min},\ U_\Sigma\left(K_B^{min}, K_B\right) \leq U_{max}$$

where R_{min} – minimal acceptable reliability of the computing system. The solution of this project task is carried out in three steps.

Step 1. Design an architecture with complete connections that meets the requirements regarding costs and reliability. The maximum computational efficiency is characterized by the architecture in which all the transceiver elements are used. In particular, the system should meet the following condition: $K_B^{min} = K_B = K_{in}$. However, because all buses are used, such architecture is also the most unreliable. Therefore, at the beginning the architecture with maximum number of channels from the range $1 \leq K_B \leq K_{in}$ is sought. The cost limit $U_\Sigma\left(K_B^{min}, K_B\right)$ should be met.

Further, the configurations with the minimal number of buses K_B^{min} from the range $K_B - 1, K_B - 2, \dots, 1$ are gradually determined. Reliability is calculated for each of them. The searching procedure is continued until a configuration meeting the reliability condition is found. Among them, one is selected, and it should satisfy all other requirements. This step of design procedure is shown as Algorithm 3. If all the requirements are met, optimal number of buses K_B^{opt} and maximum value of efficiency D_C^{max} are returned. Otherwise (i.e. $K_B^{opt} = 0$), appropriate connection system does not exist.

Algorithm 3. The first step of designing computationally efficient system

```
input: $K_{in}$, $R_{min}$, $t_{max}$, $U_{max}$
for $l \leftarrow 0$ to $K_{in}$
  if $(U_\Sigma(l) \leq U_{max})$
    exit for
  end if
end for
if $(l > K_{in})$
  return $\emptyset$
end if
$D_C^{max} \leftarrow 0$, $K_B^{opt} \leftarrow 0$
for $K_B \leftarrow l$ to 1
  if $(R(K_B) \geq R_{min} \wedge t(K_B) \leq t_{max})$
    if $(D_C(K_B) > D_C^{max})$
      $D_C^{max} \leftarrow D_C(K_B)$
      $K_B^{opt} \leftarrow K_B$
    end if
  end if
end for
return $\left(K_B^{opt}, D_C^{max}\right)$
```

Step 2. A configuration with complete connections is required. It should meet the requirements for total costs U_Σ and reliability R. Note that the architecture with partial connections $(K_B > K_{in})$ is usually unacceptable due to the construction cost. Therefore, determining architecture with partial connections ensuring redundancy of transceiver devices is possible only if the architecture with complete connections $(K_B = K_{in})$ satisfies cost limit. The basis for further considerations is the fact that the architecture with partial connections with $K_B = \lceil 3K_{in}/2 \rceil$ is characterized by maximum performance and cost as well as minimal latency and reliability. Therefore, for the implementation of configurations that meet the cost and reliability constraints, structures are created sequentially for which number of channels K_B is respectively equal to: $\lceil 3K_{in}/2 \rceil, \lceil 3K_{in}/2 \rceil - 1, \dots, K_{in} + 1$. For each variant, the cost and reliability are determined. The given process is continued until the configuration is defined meeting both criteria at the same time. This step is described as Algorithm 4. According to algorithm, system can consist of complete or partial connections. In the case, when requirements cannot be met, construction of the system is impossible.

Algorithm 4. The second step of designing computationally efficient system

```
input: $K_{in}$, $R_{min}$, $U_{max}$, $t_{max}$, $D_C^{min}$
for $l \leftarrow 1\ to\ K_B^{max}$
  if $(l \le K_{in})$
    if $\left(D_C(l) \ge D_C^{min} \wedge t(l) \le t_{max}\right)$
      for $k \leftarrow \lceil 3K_{in}/2 \rceil\ to\ K_{in}$
        if $(U_\Sigma(l) \le U_{max} \wedge R(l) \ge R_{min})$
          return $partialConnections$
        end if
      end for
      exit for
    end if
  else
    return $completeConnections$
  end if
end for
return $\emptyset$
```

D. Task 3: Minimization of latency

Specify the minimum number of K_B^{min} buses and the method of connecting computational nodes to them ensuring minimal latency with additional requirements: total cost U_Σ of its technical devices, reliability R and minimal computational efficiency D_C.

Task 3 is essentially the task of building a computing system with a minimum communication delay. The optimization task can be written as:

$$t\left(K_B^{min}, K_B\right) \to \min$$

with following restrictions:

$$1 \le K_B^{min} \le K_B\,,\ D_C\left(K_B^{min}, K_B\right) \ge D_C^{min}\,,$$

$$R\left(k_B^{min}, K_B\right) \ge R_{min}\,,\ U_\Sigma\left(K_B^{min}, K_B\right) \le U_{max}\,,$$

where: R_{min} – minimum acceptable reliability of the computational system.

Similar as in the previous one, task 3 is solved in three steps. The first one is based on searching for an architecture with complete connections, meeting cost and reliability constraints, characterized by minimal communication delay. The second is to search for an architecture with partial connections that meets above requirements while the third is to choose the architecture with highest efficiency.

IV. SIMULATION TESTS

The research was conducted in two independent areas. The first included theoretical studies based on models prepared and verified for this purpose. The super-microcomputer will be based on Raspberry devices connected via a multi-channel bus. For reconfiguration, transceiver devices with fixed channels were used and tuned in various configurations. The research has shown that it is possible to ensure a linear increase in the computing power of the system in a very wide range, depending only on the number of available logical buses and the possibility of their management by BCU. The studies simulated a parallel solution to classical tasks in the field of algebra, set theory and graph theory. For example, double reduction in solving time for a system of linear homogeneous Diophantine equations using the Contejean-Devie algorithm occurred after addition of 4 additional support servers. Similar results were obtained for the Pottier and Demenjoud methods. Empirical verification of modelling results was not possible now. The research confirmed the author's expectations regarding the flexibility of reconfiguration. For a dynamically changing assortment of short tasks, the computational efficiency obtained in parallel system (i.e. using support servers) was smaller than expected. For this reason, the simulations indicate a relatively long reconfiguration time using tunable transceiver devices.

The second analyzed architecture is a real commercial IT system designed to detect attacks on availability in a heterogeneous system containing both classic PC computers as well as tablets and smartphones. The research was conducted in *Dominet* corporation and has shown that in the case of an attack or the occurrence of bursty traffic, remote Raspberry modules, thanks to flexibility of connection system, can effectively perform both of their functions: a network traffic analyzer and an additional computational element. Threat detection was performed using among others AI and graph theory. Research has also shown that supporting servers are required only at the learning stage of neural network. Other system tasks are performed within an acceptable time. It was also found that adding less than 5-6 servers to the system is pointless. When solving learning tasks and searching for a perfect match in a n-partite graph, saturation of 1Gb communication channels followed the inclusion of a dozen or more support servers. The results of the experiments showed that the usefulness of the electrical system implementation to solve typical computational tasks requiring parallelism is satisfactory, but worse than the possibilities offered by optical solutions. Available reconfiguration options are small in relation to the optical environment.

V. CONCLUSIONS AND FURTHER WORK

The approach presented in this work was used to create a methodology for designing multi-channel bus connections addressed to the communication service of heterogeneous distributed system. In contrast to existing methodologies [23], which focus on ensuring a certain level of bandwidth in entire computational system omitting reliability parameters, the proposed methodology allows to determine the architecture of connections characterized by:

a. maximum reliability with minimum acceptable efficiency and maximum acceptable latency in the connection network;
b. minimum latency with a minimum acceptable reliability and efficiency;
c. maximum computational efficiency with a defined acceptable level of reliability and latency.

The maximum construction costs are limited for each of the solutions. The construction of a distributed system uses buses with complete and partial connections, both flat and hierarchical.

Further research will focus on the formalization of the specific architecture selection from a set of acceptable solutions, considering the incompleteness of information necessary in decision-making process. For this purpose, the connection network will be presented in a form of n-partite hypergraph. The solution assessment will be multi-criteria. The set of them will be flexible and its selection will depend on the designer's needs, in particular on the nature of future operation of the connection network.

Based on hypergraph model, a set of $A(a)$ acceptable solutions will be considered. For each of them, the following quality indicators will be defined:

1. The first optimization criterion applies to computational efficiency $\Phi_1(a) = \max_{a\in A} \min_{e\in E_a} \omega(e)$, where E_a – a set of hypergraph edges belonging to the solution a. Maximized of minimum level of performance (computational or communication) of the system will be considered;
2. The second optimization criterion applies to communication latency: $\Phi_2(a) = \min \sum_{e\in E_a} \xi(e)$ ensuring the search of the connection network with a minimum total latency. For systems with different importance, the value $\xi(e)$ of the expected latency change will be scaled to the node's priority;
3. The reliability criterion: $\Phi_3(a) = \max \sum_{e\in E_a} \psi(e)$. As in the case of latency criterion Φ_2, this criterion ensures the search for the network architecture with maximum total reliability.

The possibilities of this method are not limited to the summation criterion in min or max form. To evaluate the quality of the final solution, any methods of convolution can be used, including methods that consider weights of individual subparameters.

Partial criteria will be combined using the following function: $\Phi(a) = \left(\Phi_1(a), \Phi_2(a), \Phi_3(a)\right)$. The multi-criteria objective function $\Phi(a)$ determines in the set of acceptable solutions A the Pareto set A^p containing Pareto solutions a^p. If two solutions $a_1, a_2 \in A$ of the vector objective function $\Phi(a)$ are equivalent, then from the set A^p a full set of alternatives will be extracted, which is, in fact, the maximum system of vector Pareto optimizations.

Acknowledgment

The work was realized as a part of fundamental research financed by the Ministry of Science and Higher Education, grant no. 16.16.110.663, task 8.

References

[1] R. Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, Hoboken, New Jersey, USA: Wiley & Sons, 1991.

[2] N. J. Gunther, Analyzing Computer System Performance with Perl::PDQ, Heidelberg: Springer-Verlag, Germany, 2005.

[3] P. J. Fortier and H. E. Michel, Computer Systems Performance Evaluation and Prediction, Amsterdam, NL: Digital Press, 2003.

[4] B. Greg, Systems Performance. Enterprise and the Cloud, Upper Saddle River, NJ, USA: Prentice Hall, 2014.

[5] A. Oppermann, M. Esche, F. Thiel, J.-P. Seifert, "Secure Cloud Computing: Risk Analysis for Secure Cloud Reference Architecture in Legal Metrology," in Proceedings of the Federated Conference on Computer Science and Information Systems, pp. 593-602, 2018, DOI: 10.15439/2018F226

[6] A. D. Malony, S. Biersdorff, S. Shende, H. Jagode, S. Tomov, G. Juckeland, R. Dietrich, D. Poole and C. Lamb, "Parallel Performance Measurement of Heterogeneous Parallel Systems with GPUs," in International Conference on Parallel Processing, 2011.

[7] M. U. Asharif, F. A. Eassa, A. A. Albeshri and A. Algarni, "Performance and Power Efficient Massive Parallel Computational Model for HPC Heterogeneous Exascale," IEEE Access, vol. 6, pp. 23095-23107, 2018.

[8] K. Vanishree and M. Purnaprajna, "Work in Progress: Performance Modeling for Data Distribution in Heterogeneous Computing Systems," in International Conference on Compilers, Architectures and Synthesis for Embedded Systems (CASES), Torino, Italy, 2018.

[9] M. Hajder, H. Loutskii and W. Stręciwilk, Informatyka. Wirtualna podróż w świat systemów i sieci komputerowych, M. Hajder, Red., Rzeszów: Wydawnictwo Wyższej Szkoły Informatyki i Zarządzania, 2002.

[10] T. Sterling, M. Anderson and M. Brodowicz, High Performance Computing. Modern Systems and Practices, Cambridge, MA, USA: Morgan Kaufmann, 2018.

[11] S. Bhowmik, Cloud Computing, Cambridge: Cambridge University Press, 2017.

[12] M. Hajder and M. Kiełbus, „Matematyczny model opóźnień w sieci z komutacją pakietów," w XV Konferencja Sieci i Systemy Informatyczne, Łódź, 2007.

[13] C. Avin and S. Schmid, "Toward demand-aware networking: a theory for self-adjusting networks," SIGCOMM Comput. Commun. Rev., vol. 48, no. 5, pp. 31-40, 2018.

[14] P. Hajder and L. Rauch, "Reconfiguration of the Multi-channel Communication System with Hierarchical Structure and Distributed Passive Switching," in ICCS 2019, Part II, LNCS 11537 proceedings, 2019.

[15] M. Hajder, Reconfigurable multichannel network communication systems architecture, Rzeszow: Wydawnictwo Wyższej Szkoły Informatyki i Zarządzania w Rzeszowie, 2018.

[16] M. Hajder and M. Bolanowski, "Connectivity Analysis in the Computational Systems with Distributed Communications in the Multichanel Environment," POLISH JOURNAL OF ENVIRONMENTAL STUDIES, vol. 17, no. 2A, pp. 14-18, 2008.

[17] M. Hajder, M. Bolanowski and L. Byczkowska-Lipińska, "A set of connection network synthesis basis on the linear diophantine constraints solution in area {0,1}," in Ogólnopolskie Warsztaty Doktoranckie. Materiały konferencji, Nałęczów, 2008.

[18] M. Hajder, Methods and facilities increasing effectiveness of designing distributed systems based on multichannels. Dissertation for the degree Doctor of Technical Science, Kyiv: National Technical University of Ukraine "Kyiv Polytechnic Institute", 2006.

[19] J. Kunegis, "Exploiting the structure of bipartite graphs for algebraic and spectral graph theory," Internet Mathematics, vol. 11, no. 3, pp. 201-321, 2015.

[20] M. Hajder, J. Kolbusz and T. Bartczak, "Undirected graph algebra application for formalization description of information flows," in Human System Interaction (HSI), 2013 The 6th International Conference, Kraków, 2013.

[21] M. Hajder, H. Loutskii and W. Stręciwilk, Informatyka. Wirtualna podróż w świat systemów i sieci komputerowych, Rzeszów: Wydawnictwo Wyższej Szkoły Informatyki i Zarządzania w Rzeszowie, 2002.

[22] M. Hajder and P. Dymora, "A novel approach to fault tolerant multichannel networks designing problems," ANNALES UNIVERSITATIS MARIAE CURIE-SKŁODOWSKA, SECTIO AI: INFORMATICA, no. 1, pp. 66-77, 2011.

[23] M. Хайдер, "Множественный подход в системах каналов связи," Вісник Національного технічного університету України "Київський політехнічний інститут". Інформатика, управління та обчислювальна техніка, vol. 41, pp. 120-132, 2004.

DOI: 10.15439/2019F338
ISSN 2300-5963 ACSIS, Vol. 19

Alignment for Rooted Labeled Caterpillars

Yoshiyuki Ukita
Graduate School of Computer Science and Systems Engineering
Kyushu Institute of Technology
Kawazu 680-4, Iizuka 820-8502, Japan
Email: o231014y@mail.kyutech.jp

Takuya Yoshino, Kouichi Hirata*
Department of Artificial Intelligence
Kyushu Institute of Technology
Kawazu 680-4, Iizuka 820-8502, Japan
Email: {yoshino,hirata}@ai.kyutech.ac.jp

***Abstract*—A *rooted labeled caterpillar* (*caterpillars*, for short) is a rooted labeled tree transformed to a rooted path after removing all the leaves in it. In this paper, we design the algorithm to compute the *alignment* distance between caterpillars in $O(h^2\lambda^3)$ time under the general cost function and in $O(h^2\lambda)$ time under the unit cost function, where h is the maximum height and λ is the maximum number of leaves in caterpillars.**

I. Introduction

COMPARING tree-structured data such as HTML and XML data for web mining or RNA and glycan data for bioinformatics is one of the important tasks for data mining. The most famous distance measure [2] between *rooted labeled unordered trees* (*trees*, for short) is the *edit distance* [10]. The edit distance is formulated as the minimum cost of *edit operations*, consisting of a *substitution*, a *deletion* and an *insertion*, applied to transform a tree to another tree. It is known that the edit distance is always a metric and coincides with the minimum cost of *Tai mappings* [10].

Unfortunately, the problem of computing the edit distance between trees is MAX SNP-hard [15]. This statement also holds even if trees are binary or the maximum height of trees is at most 3 [1], [4].

A *caterpillar* (*cf.* [3]) is a tree transformed to a rooted path after removing all the leaves in it. Whereas the caterpillars are very restricted and simple, there are some cases containing many caterpillars in real dataset, see Table I in Appendix. Recently, Muraka *et al.* [8] have proposed the algorithm to compute the edit distance between caterpillars in $O(h^2\lambda^3)$ time under the general cost function and in $O(h^2\lambda)$ time under the unit cost function, where h is the maximum height and λ is the maximum number of leaves in caterpillars[1]. They have also introduced the efficient comparable distances to approximate the edit distance between caterpillars [9].

An *alignment distance* is an alternative distance measure between trees, introduced by Jiang *et al.* [5]. The alignment distance between two trees is formulated as the minimum cost of possible *alignments* (as trees) obtained by first inserting nodes labeled with spaces into two trees so that the resulting trees have the same structure and then overlaying them. In operational, the alignment distance is regarded as an edit distance such that every insertion precedes to deletions. Hence, the alignment distance between trees is not always equal to the edit distance and regarded as a variation of the edit distance. Furthermore, Kuboyama [6] has shown that the alignment distance coincides with the minimum cost of *less-constrained mappings* [7], which is the restriction of the Tai mapping.

As same as the edit distance, the problem of computing the alignment distance between trees is also MAX SNP-hard [5]. On the other hand, it is tractable if the degrees are bounded by some constant [5]. Since a caterpillar is not a bounded-degree tree, it is still open whether or not the problem of computing the alignment distance is tractable,

In this paper, first we point out that there exists a pair of caterpillars whose minimum cost less-constrained mapping is not an isolated-subtree mapping and whose minimum cost Tai mapping is not a less-constrained mapping. Then, we can apply the algorithm of computing neither the isolated-subtree distance or its variations [12], [13], [14], [16] nor the edit distance [8] to compute the alignment distance between caterpillars.

Next, we design the algorithm to compute the alignment distance between caterpillars in $O(h^2\lambda^3)$ time under the general cost function and in $O(h^2\lambda)$ time under the unit cost function. Here, it is necessary to adopt the edit distance for multisets (*cf.*, [9]) to compute the alignment distance between sets of leaves. Furthermore, as same as the edit distance [8], we point out the structural restriction of caterpillars provides the limitation of tractable computing of the alignment distance for unordered trees.

II. Preliminaries

In this section, we prepare the notions necessary to discuss the later sections.

A *tree* T is a connected graph (V, E) without cycles, where V is the set of vertices and E is the set of edges. We denote V and E by $V(T)$ and $E(T)$. The *size* of T is $|V|$ and denoted by $|T|$. We sometime denote $v \in V(T)$ by $v \in T$. We denote an empty tree $(\emptyset, \emptyset)$ by $\emptyset$. A *rooted tree* is a tree with one node r chosen as its *root*. We denote the root of a rooted tree T by $r(T)$.

Let T be a rooted tree such that $r = r(T)$ and $u, v, w \in T$. We denote the unique path from r to v, that is, the tree

*The author would like to express thanks for support by Grant-in-Aid for Scientific Research 17H00762, 16H02870 and 16H01743 from the Ministry of Education, Culture, Sports, Science and Technology, Japan.

[1] This time complexity is different from the result in [8], because it contains some errors. See Appendix.

(V', E') such that $V' = \{v_1, \ldots, v_k\}$, $v_1 = r$, $v_k = v$ and $(v_i, v_{i+1}) \in E'$ for every i $(1 \leq i \leq k-1)$, by $UP_r(v)$.

The *parent* of $v (\neq r)$, which we denote by $par(v)$, is its adjacent node on $UP_r(v)$ and the *ancestors* of $v (\neq r)$ are the nodes on $UP_r(v) - \{v\}$. We say that u is a *child* of v if v is the parent of u and u is a *descendant* of v if v is an ancestor of u. We denote the set of children of v by $ch(v)$. We call a node with no children a *leaf* and denote the set of all the leaves in T by $lv(T)$.

The *degree* of v, denoted by $d(v)$, is the number of children of v, and the *degree* of T, denoted by $d(T)$, is $\max\{d(v) \mid v \in T\}$. The *height* of v, denoted by $h(v)$, is $\max\{|UP_v(w)| \mid w \in lv(T[v])\}$, and the *height* of T, denoted by $h(T)$, is $\max\{h(v) \mid v \in T\}$.

We use the ancestor orders $<$ and $\leq$, that is, $u < v$ if v is an ancestor of u and $u \leq v$ if $u < v$ or $u = v$. We say that w is the *least common ancestor* of u and v, denoted by $u \sqcup v$, if $u \leq w$, $v \leq w$ and there exists no node $w' \in T$ such that $w' \leq w$, $u \leq w'$ and $v \leq w'$. Let T be a rooted tree (V, E) and v a node in T. A *complete subtree of T at v*, denoted by $T[v]$, is a rooted tree $T' = (V', E')$ such that $r(T') = v$, $V' = \{u \in V \mid u \leq v\}$ and $E' = \{(u, w) \in E \mid u, w \in V'\}$.

We say that u is *to the left of* v in T if $pre(u) \leq pre(v)$ for the preorder number *pre* in T and $post(u) \leq post(v)$ for the postorder number *post* in T. We say that a rooted tree is *ordered* if a left-to-right order among siblings is given; *unordered* otherwise. We say that a rooted tree is *labeled* if each node is assigned a symbol from a fixed finite alphabet Σ. For a node v, we denote the label of v by $l(v)$, and sometimes identify v with $l(v)$. In this paper, we call a rooted labeled unordered tree a *tree* simply. Furthermore, we call a set of trees a *forest*.

As the restricted form of trees, we introduce a *rooted labeled caterpillar* (*caterpillar*, for short) as follows, which this paper mainly deals with.

Definition 1 (Caterpillar (cf., [3])): We say that a tree is a *caterpillar* if it is transformed to a rooted path after removing all the leaves in it. For a caterpillar C, we call the remained rooted path a *backbone* of C and denote it by $bb(C)$.

It is obvious that $r(C) = r(bb(C))$ and $V(C) = bb(C) \cup lv(C)$ for a caterpillar C, that is, every node in a caterpillar is either a leaf or an element of the backbone.

Next, we introduce an *edit distance* and a *Tai mapping* between trees.

Definition 2 (Edit operations for trees [10]): The *edit operations* of a tree T are defined as follows, see Figure 1.

1) *Substitution*: Change the label of the node v in T.
2) *Deletion*: Delete a node v in T with parent v', making the children of v become the children of v'. The children are inserted in the place of v as a subset of the children of v'. In particular, if v is the root in T, then the result applying the deletion is a forest consisting of the children of the root.
3) *Insertion*: The complement of deletion. Insert a node v as a child of v' in T making v the parent of a subset of the children of v'.

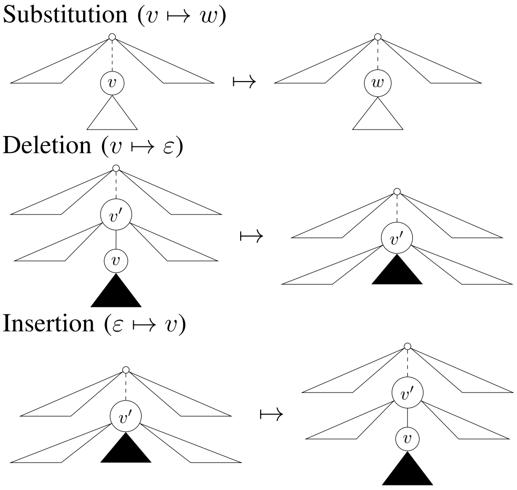

Substitution ($v \mapsto w$)
v
$\mapsto$
w
Deletion ($v \mapsto \varepsilon$)
v'
v
$\mapsto$
v'
Insertion ($\varepsilon \mapsto v$)
v'
$\mapsto$
v'
v

Fig. 1. Edit operations for trees.

Let $\varepsilon \notin \Sigma$ denote a special *blank* symbol and define $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$. Then, we represent each edit operation by $(l_1 \mapsto l_2)$, where $(l_1, l_2) \in (\Sigma_\varepsilon \times \Sigma_\varepsilon - \{(\varepsilon, \varepsilon)\})$. The operation is a substitution if $l_1 \neq \varepsilon$ and $l_2 \neq \varepsilon$, a deletion if $l_2 = \varepsilon$, and an insertion if $l_1 = \varepsilon$. For nodes v and w, we also denote $(l(v) \mapsto l(w))$ by $(v \mapsto w)$.

We define a *cost function* $\gamma : (\Sigma_\varepsilon \times \Sigma_\varepsilon \setminus \{(\varepsilon, \varepsilon)\}) \mapsto \mathbf{R}^+$ on pairs of labels. For $(v, w) \in V(T_1) \times V(T_2)$, we also denote $\gamma(l(v), l(w))$ by $\gamma(v, w)$ simply.

We often constrain a cost function γ to be a *metric*, that is, $\gamma(l_1, l_2) \geq 0$, $\gamma(l_1, l_2) = 0$ iff $l_1 = l_2$, $\gamma(l_1, l_2) = \gamma(l_2, l_1)$ and $\gamma(l_1, l_3) \leq \gamma(l_1, l_2) + \gamma(l_2, l_3)$. In particular, we call the cost function that $\gamma(l_1, l_2) = 1$ if $l_1 \neq l_2$ a *unit cost function*.

Definition 3 (Edit distance for trees [10]): For a cost function γ, the *cost* of an edit operation $e = l_1 \mapsto l_2$ is given by $\gamma(e) = \gamma(l_1, l_2)$. The *cost* of a sequence $E = e_1, \ldots, e_k$ of edit operations is given by $\gamma(E) = \sum_{i=1}^{k} \gamma(e_i)$. Then, an *edit distance* $\tau_{\text{TAI}}(T_1, T_2)$ between trees T_1 and T_2 is defined as follows:

$$\tau_{\text{TAI}}(T_1, T_2) = \min \left\{ \gamma(E) \;\middle|\; \begin{array}{l} E \text{ is a sequence} \\ \text{of edit operations} \\ \text{transforming } T_1 \text{ to } T_2 \end{array} \right\}.$$

Definition 4 (Tai mapping [10]): Let T_1 and T_2 be trees. We say that a triple (M, T_1, T_2) is a *Tai mapping* (a *mapping*, for short) from T_1 to T_2 if $M \subseteq V(T_1) \times V(T_2)$ and every pair (v_1, w_1) and (v_2, w_2) in M satisfies the following conditions.

1) $v_1 = v_2$ iff $w_1 = w_2$ (one-to-one condition).
2) $v_1 \leq v_2$ iff $w_1 \leq w_2$ (ancestor condition).

We will use M instead of (M, T_1, T_2) when there is no confusion denote it by $M \in \mathcal{M}_{\text{TAI}}(T_1, T_2)$.

Let M be a mapping from T_1 to T_2. Let I_M and J_M be the sets of nodes in T_1 and T_2 but not in M, that is, $I_M = \{v \in T_1 \mid (v, w) \notin M\}$ and $J_M = \{w \in T_2 \mid (v, w) \notin M\}$. Then, the *cost* $\gamma(M)$ of M is given as follows.

$$\gamma(M) = \sum_{(v,w) \in M} \gamma(v, w) + \sum_{v \in I_M} \gamma(v, \varepsilon) + \sum_{w \in J_M} \gamma(\varepsilon, w).$$

Trees T_1 and T_2 are *isomorphic*, denoted by $T_1 \equiv T_2$, if there exists a mapping $M \in \mathcal{M}_{\rm TAI}(T_1, T_2)$ such that $I_M = J_M = \emptyset$ and $\gamma(M) = 0$.

Theorem 1 (Tai [10]): $\tau_{\rm TAI}(T_1, T_2) = \min\{\gamma(M) \mid M \in \mathcal{M}_{\rm TAI}(T_1, T_2)\}$.

III. Alignment Distance

In this section, we introduce the alignment distance and characterize it by using the variation of Tai mappings.

Definition 5 (Alignment [5]): Let T_1 and T_2 be trees. Then, an *alignment* between T_1 and T_2 is a tree $\mathcal{T}$ obtained by the following steps.

1) Insert new nodes labeled by ε into T_1 and T_2 so that the resulting trees T_1' and T_2' are isomorphic with ignoring labels and $l(\phi(v)) \neq \varepsilon$ whenever $l(v) = \varepsilon$ for an isomorphism ϕ from T_1' to T_2' and every node $v \in T_1'$.
2) Set $\mathcal{T}$ to a tree T_1' obtained by relabeling a label $l(v)$ for every node $v \in T_1'$ with $(l(v), l(\phi(v)))$. (Note that $(\varepsilon, \varepsilon) \notin \mathcal{T}$.)

Let $\mathcal{A}(T_1, T_2)$ denote the set of all possible alignments between trees T_1 and T_2.

For a cost function γ, the *cost* of an alignment $\mathcal{T}$, denoted by $\gamma(\mathcal{T})$, is the sum of the costs of all labels in $\mathcal{T}$.

Definition 6 (Alignment distance [5]): Let T_1 and T_2 be trees and γ a cost function. Then, an *alignment distance* $\tau_{\rm ALN}(T_1, T_2)$ between T_1 and T_2 is defined as follows.

$$\tau_{\rm ALN}(T_1, T_2) = \min\{\gamma(\mathcal{T}) \mid \mathcal{T} \in \mathcal{A}(T_1, T_2)\}.$$

Also we call an alignment between T_1 and T_2 with the minimum cost an *optimal alignment* and denote it by $\mathcal{A}^*(T_1, T_2)$.

The notion of the alignment can be easily extended to forests. The only change is that it is now possible to insert a node (as the root) of trees in the forest. We denote the set of all possible alignments between forests F_1 and F_2 by $\mathcal{A}(F_1, F_2)$ and an optimal alignment by $\mathcal{A}^*(F_1, F_2)$.

Example 1: For two caterpillars C_1 and C_2 illustrated in Figure 2, $\mathcal{A}^*(C_1, C_2)$ is the optimal alignment between C_1 and C_2. Also, for two caterpillars C_3 and C_4 illustrated in Figure 2, $\mathcal{A}^*(C_3, C_4)$ is the optimal alignment between C_3 and C_4. Under the unit cost function, it holds that $\tau_{\rm ALN}(C_1, C_2) = 3$ and $\tau_{\rm ALN}(C_3, C_4) = 3$.

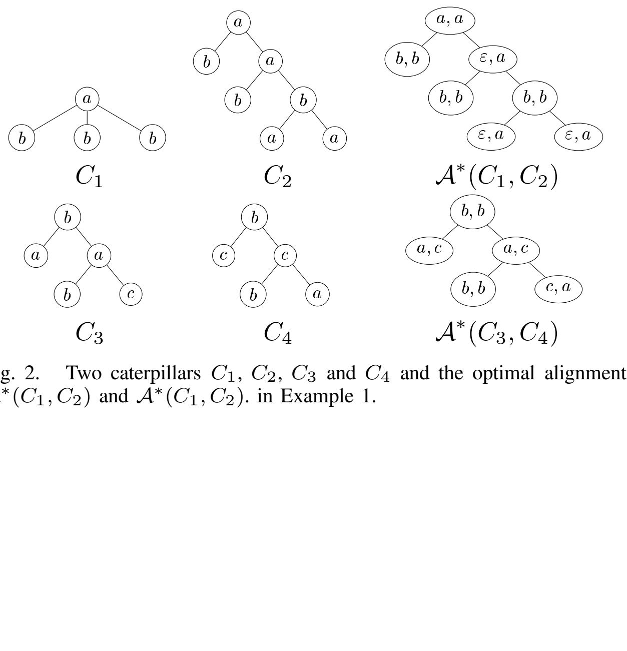


Fig. 2. Two caterpillars C_1, C_2, C_3 and C_4 and the optimal alignments $\mathcal{A}^*(C_1, C_2)$ and $\mathcal{A}^*(C_1, C_2)$. in Example 1.

Next, we introduce the variations of Tai mappings, including the mapping characterizing the alignment distance.

Definition 7 (Variations of Tai mapping): Let T_1 and T_2 be trees and $M \in \mathcal{M}_{\rm TAI}(T_1, T_2)$.

1) We say that M is a *less-constrained mapping* [7], denoted by $M \in \mathcal{M}_{\rm LESS}(T_1, T_2)$, if M satisfies the following condition for every $(v_1, w_1), (v_2, w_2), (v_3, w_3) \in M$:
$$(v_1 \sqcup v_2 < v_1 \sqcup v_3) \Longrightarrow (w_2 \sqcup w_3 = w_1 \sqcup w_3).$$
Also we define a *less-constrained distance* $\tau_{\rm LESS}(T_1, T_2)$ as the minimum cost of all the less-constrained mappings, that is:
$$\tau_{\rm LESS}(T_1, T_2) = \min\{\gamma(M) \mid M \in \mathcal{M}_{\rm LESS}(T_1, T_2)\}.$$
2) We say that M is an *isolated-subtree mapping* [11] (or a *constrained mapping* [14]), denoted by $M \in \mathcal{M}_{\rm ILST}(T_1, T_2)$, if M satisfies the following condition for every $(v_1, w_1), (v_2, w_2), (v_3, w_3) \in M$:
$$(v_3 < v_1 \sqcup v_2) \iff (w_3 < w_1 \sqcup w_2).$$
Also we define an *isolated-subtree distance* $\tau_{\rm ILST}(T_1, T_2)$ as the minimum cost of all the isolated-subtree mappings, that is:
$$\tau_{\rm ILST}(T_1, T_2) = \min\{\gamma(M) \mid M \in \mathcal{M}_{\rm ILST}(T_1, T_2)\}.$$

Theorem 2: Let T_1 and T_2 be trees, where $n = \max\{|T_1|, |T_2|\}$ and $d = \min\{d(T_1), d(T_2)\}$.

1) It holds that $\tau_{\rm ALN}(T_1, T_2) = \tau_{\rm LESS}(T_1, T_2)$ [6]. Also it holds that $\tau_{\rm TAI}(T_1, T_2) \leq \tau_{\rm ALN}(T_1, T_2) \leq \tau_{\rm ILST}(T_1, T_2)$ but the equations always do not hold (*cf.*, [5], [6], [14]).
2) The problem of computing $\tau_{\rm TAI}(T_1, T_2)$ is MAX SNP-hard [15]. This statement holds even if both T_1 and T_2 are binary, the maximum height of T_1 and T_2 is at most 3 or the cost function is the unit cost function [1], [4].
3) The problem of computing $\tau_{\rm ALN}(T_1, T_2)$ is MAX SNP-hard. On the other hand, if the degrees of T_1 and T_2 are bounded by some constants, then we can compute $\tau_{\rm ALN}(T_1, T_2)$ in polynomial time with respect to n [5].
4) We can compute $\tau_{\rm ILST}(T_1, T_2)$ in $O(n^2 d)$ time (*cf.*, [12]).

Example 2: Consider two caterpillars C_1 and C_2 in Figure 2 in Example 1 and assume the unit cost function. Then, M_1 and M_2 illustrated in Figure 3 are the minimum cost mappings in $\mathcal{M}_{\rm LESS}(C_1, C_1) (= \mathcal{M}_{\rm TAI}(C_1, C_2))$ and $\mathcal{M}_{\rm ILST}(C_1, C_2)$. Here, it holds that $M_1 \notin \mathcal{M}_{\rm ILST}(C_1, C_2)$. Then, it holds that $\tau_{\rm TAI}(C_1, C_2) = \tau_{\rm ALN}(C_1, C_2) = 3 < 5 = \tau_{\rm ILST}(C_1, C_2)$.

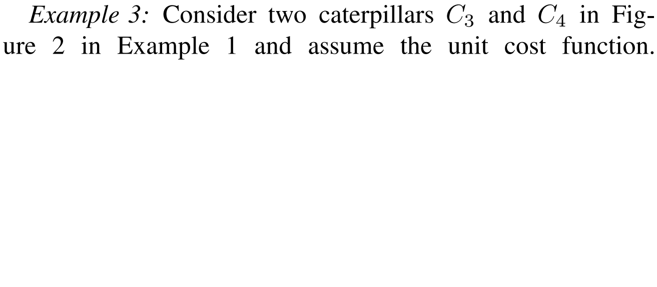


Fig. 3. The minimum cost mappings $M_1 \in \mathcal{M}_{\rm LESS}(C_1, C_2)$ and $M_2 \in \mathcal{M}_{\rm ILST}(C_1, C_2)$ in Example 2.

Example 3: Consider two caterpillars C_3 and C_4 in Figure 2 in Example 1 and assume the unit cost function.

Then, M_3 and M_4 illustrated in Figure 4 are the minimum cost mappings in $\mathcal{M}_{\text{TAI}}(C_3, C_4)$ and $\mathcal{M}_{\text{LESS}}(C_3, C_4)$. Here, it holds that $M_3 \notin \mathcal{M}_{\text{LESS}}(C_3, C_4)$. Hence, it holds that $\tau_{\text{TAI}}(C_3, C_4) = 2 < 3 = \tau_{\text{ALN}}(C_3, C_4)$.

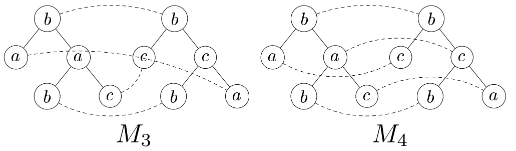


Fig. 4. The minimum cost mappings $M_3 \in \mathcal{M}_{\text{TAI}}(C_3, C_4)$ and $M_4 \in \mathcal{M}_{\text{LESS}}(C_3, C_4)$ in Example 3.

Example 2 shows that there exists a pair of caterpillars whose minimum cost less-constrained mapping is not an isolated-subtree mapping. Then, we cannot use the algorithm to compute the isolated-subtree distance between caterpillars [12], [13], [14], [16] to compute their alignment distance. Also Example 3 shows that there exists a pair of caterpillars whose minimum cost Tai mapping is not a less-constrained mapping. Then, we cannot use the algorithm to compute the edit distance between caterpillars [8] to compute their alignment distance. Furthermore, it still remains open whether or not Theorem 2.3 holds for caterpillars.

Hence, in the next section, we discuss the problem of computing the alignment distance between caterpillars.

IV. The Algorithm of Computing Alignment Distance between Caterpillars

In this section, we design the algorithm to compute the alignment distance τ_{ALN} between caterpillars.

A. Edit distance for multisets

In order to compute the edit distance between the sets of leaves, it is necessary to introduce an edit distance for *multisets* on labels occurring in the set of leaves. Then, we prepare the notions of the edit distance for multisets according to [9].

A *multiset* on an alphabet Σ is a mapping $S : \Sigma \to \mathbf{N}$. For a multiset S on Σ, we say that $a \in \Sigma$ is an *element* of S if $S(a) > 0$ and denote it by $a \in S$ (like as a standard set). The *cardinality* of S, denoted by $|S|$, is defined as $\sum_{a \in \Sigma} S(a)$.

Definition 8 (Edit operations for multisets): Let $a, b \in \Sigma$ such that $S(a) > 0$ and $a \neq b$. Then, a *substitution* $(a \mapsto b)$ operates $S(a)$ to $S(a) - 1$ and $S(b)$ to $S(b) + 1$, a *deletion* $(a \mapsto \varepsilon)$ operates $S(a)$ to $S(a) - 1$ and an *insertion* $(\varepsilon \mapsto b)$ operates $S(b)$ to $S(b) + 1$.

Also we assume a cost function γ as in Section II.

Definition 9 (Edit distance for multisets): Let S_1 and S_2 be multisets on Σ and γ a cost function. Then, an *edit distance* $\mu(S_1, S_2)$ between S_1 and S_2 is defined as follows.

$$\mu(S_1, S_2) = \min \left\{ \gamma(E) \;\middle|\; \begin{array}{l} E \text{ is a sequence} \\ \text{of edit operations} \\ \text{transforming } S_1 \text{ to } S_2 \end{array} \right\}.$$

For multisets S_1 and S_2 on Σ, we define the *difference* $S_1 \setminus S_2$ between S_1 and S_2 as a multiset satisfying that $(S_1 \setminus S_2)(a) = \max\{S_1(a) - S_2(a), 0\}$ for every $a \in \Sigma$.

Lemma 1 ([9]): Let Π_1 be the set of all the injections from S_1 to S_2 when $|S_1| \leq |S_2|$ and Π_2 the set of all the injections from S_2 to S_1 when $|S_1| > |S_2|$. Then, we can compute $\mu(S_1, S_2)$ as follows:

$$\mu(S_1, S_2) = \begin{cases} \min\limits_{\pi \in \Pi_1} \left\{ \sum\limits_{a \in S_1} \gamma(a, \pi(a)) + \sum\limits_{b \in S_2 \setminus \pi(S_1)} \gamma(\varepsilon, b) \right\}, \\ \quad \text{if } |S_1| \leq |S_2|, \\ \min\limits_{\pi \in \Pi_2} \left\{ \sum\limits_{b \in S_2} \gamma(\pi(b), b) + \sum\limits_{a \in S_1 \setminus \pi(S_2)} \gamma(a, \varepsilon) \right\}, \\ \quad \text{otherwise.} \end{cases}$$

Furthermore, if we adopt the unit cost function, then we can compute $\mu(S_1, S_2)$ as follows:

$$\mu(S_1, S_2) = \max\{|S_1 \setminus S_2|, |S_2 \setminus S_1|\}.$$

In this case, $\mu(S_1, S_2)$ coincides with a famous *bag distance* (*cf.*, [2]) between multisets S_1 and S_2.

Lemma 2 ([9]): Let $m = \max\{|S_1|, |S_2|\}$. Then, we can compute $\mu(S_1, S_2)$ in $O(m^3)$ time under the general cost function. If we adopt the unit cost function, then we can compute $\mu(S_1, S_2)$ in $O(m)$ time.

B. Recurrences

Let L be the set of leaves and C a non-leaf caterpillar. Then, every forest obtained by deleting the root from a caterpillar is one of the forms of $\{C\}$, L or $L \cup \{C\}$. As same as [8], we denote these forests by $\langle \emptyset \,|\, C \rangle$, $\langle L \,|\, \emptyset \rangle$ and $\langle L \,|\, C \rangle$, respectively. In particular, we denote an empty forest $\langle \emptyset \,|\, \emptyset \rangle$ by Φ simply.

Let $C[v]$ be a caterpillar with the root v, where $L(v)$ denotes a (possibly empty) set of leaves as the children of v and $B(v)$ denotes at most one caterpillar of the child v. Then, $C[v]$ is one of the forms in Figure 5. Furthermore, by deleting v from $C[v]$, we obtain one of the forests of $\langle \emptyset \,|\, B(v) \rangle$, $\langle L(v) \,|\, \emptyset \rangle$ and $\langle L(v) \,|\, B(v) \rangle$, respectively.

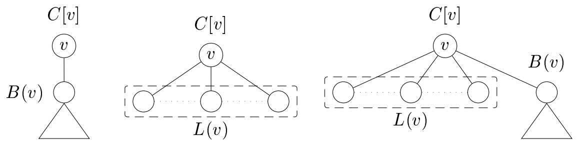


Fig. 5. The representation of a caterpillar $C[v]$.

Figure 6 illustrates the recurrences of computing the alignment distance $\tau_{\text{ALN}}(C_1[v], C_2[w])$ between two caterpillars $C_1[v]$ and $C_2[w]$. Here, we regard a set L of leaves as a multiset of labels on Σ occurring in L, which we denote by $\widetilde{L}$. Also $\delta_{\text{ALN}}(\langle L_1 \mid B_1 \rangle, \langle L_2 \mid B_2 \rangle)$ describes the alignment distance between forests $\langle L_1 \,|\, B_1 \rangle$ and $\langle L_2 \,|\, B_2 \rangle$. Furthermore, assume that the forest obtained by deleting v (*resp.*, w) from $C_1[v]$ (*resp.*, $C_2[w]$) is $\langle L_1(v) \,|\, B_1(v) \rangle$ (*resp.*, $\langle L_2(w) \,|\, B_2(w) \rangle$).

Theorem 3: The recurrences in Figure 6 are correct to compute the alignment distance $\tau_{\text{ALN}}(C_1[v], C_2[w])$ between $C_1[v]$ and $C_2[w]$.

$$\tau_{\mathrm{ALN}}(\emptyset,\emptyset) = 0. \qquad (T_0)$$

$$\tau_{\mathrm{ALN}}(C_1[v],\emptyset) = \gamma(v,\varepsilon) + \delta_{\mathrm{ALN}}(\langle L_1(v)\,|\,B_1(v)\rangle,\Phi). \qquad (T_1)$$

$$\tau_{\mathrm{ALN}}(\emptyset,C_2[w]) = \gamma(\varepsilon,w) + \delta_{\mathrm{ALN}}(\Phi,\langle L_2(w)\,|\,B_2(w)\rangle). \qquad (T_2)$$

$$\delta_{\mathrm{ALN}}(\langle L_1\,|\,C_1\rangle,\Phi) = \sum_{v\in L_1}\gamma(v,\varepsilon) + \sum_{v\in C_1}\gamma(v,\varepsilon). \qquad (F_1)$$

$$\delta_{\mathrm{ALN}}(\Phi,\langle L_2\,|\,C_2\rangle) = \sum_{w\in L_2}\gamma(\varepsilon,w) + \sum_{w\in C_2}\gamma(\varepsilon,w). \qquad (F_2)$$

$$\delta_{\mathrm{ALN}}(\langle L_1\,|\,\emptyset\rangle,\langle L_2\,|\,\emptyset\rangle) = \mu(\widetilde{L_1},\widetilde{L_2}). \qquad (F_3)$$

$$(A)\ \tau_{\mathrm{ALN}}(C_1[v],C_2[w]) = \min\left\{\begin{array}{ll} \gamma(v,w) & \\ \quad +\,\delta_{\mathrm{ALN}}(\langle L_1(v)\,|\,B_1(v)\rangle,\langle L_2(w)\,|\,B_2(w)\rangle), & (T_3) \\ \gamma(v,\varepsilon)+\tau_{\mathrm{ALN}}(B_1(v),C_2[w]) & \\ \quad +\,\delta_{\mathrm{ALN}}(\langle L_1(v)\,|\,\emptyset\rangle,\Phi), & (T_4) \\ \gamma(\varepsilon,w)+\tau_{\mathrm{ALN}}(C_1[v],B_2(w)) & \\ \quad +\,\delta_{\mathrm{ALN}}(\Phi,\langle L_2(w)\,|\,\emptyset\rangle) & (T_5) \end{array}\right\}.$$

$$(B)\ \delta_{\mathrm{ALN}}(\langle L_1\,|\,\emptyset\rangle,\langle L_2\,|\,C_2[w]\rangle) = \min\left\{\begin{array}{ll} \gamma(\varepsilon,w) & \\ \quad +\delta_{\mathrm{ALN}}(\langle L_1\,|\,\emptyset\rangle,\langle L_2\cup L_2(w)\,|\,B_2(w)\rangle), & (F_4) \\ \min\limits_{v\in L_1}\{\gamma(v,w)+\delta_{\mathrm{ALN}}(\langle L_1\setminus\{v\}\,|\,\emptyset\rangle,\langle L_2\,|\,\emptyset\rangle)\} & \\ \quad +\delta_{\mathrm{ALN}}(\Phi,\langle L_2(w)\,|\,B_2(w)\rangle) & (F_5) \end{array}\right\}.$$

$$(C)\ \delta_{\mathrm{ALN}}(\langle L_1\,|\,C_1[v]\rangle,\langle L_2\,|\,\emptyset\rangle) = \min\left\{\begin{array}{ll} \gamma(v,\varepsilon) & \\ \quad +\delta_{\mathrm{ALN}}(\langle L_1\cup L_1(v)\,|\,B_1(v)\rangle,\langle L_2\,|\,\emptyset\rangle), & (F_6) \\ \min\limits_{w\in L_2}\{\gamma(v,w)+\delta_{\mathrm{ALN}}(\langle L_1\,|\,\emptyset\rangle,\langle L_2\setminus\{w\}\,|\,\emptyset\rangle)\} & \\ \quad +\delta_{\mathrm{ALN}}(\langle L_1(v)\,|\,B_1(v)\rangle,\Phi) & (F_7) \end{array}\right\}.$$

$$(D)\ \delta_{\mathrm{ALN}}(\langle L_1\,|\,C_1[v]\rangle,\langle L_2\,|\,C_2[w]\rangle) = \delta_{\mathrm{ALN}}(\langle L_1\,|\,\emptyset\rangle,\langle L_2\,|\,\emptyset\rangle) + \tau_{\mathrm{ALN}}(C_1[v],C_2[w]). \qquad (F_8)$$

Fig. 6. The recurrences of computing the alignment distance $\tau_{\mathrm{ALN}}(C_1[v],C_2[w])$ between $C_1[v]$ and $C_2[w]$.

Proof: The recurrences of (T_0), (T_1), (T_2), (F_1), (F_2) and (F_3) are obvious.

First, consider the recurrences for τ_{ALN}. Let $\mathcal{T}$ be the optimal alignment (tree) $\mathcal{A}^*(C_1[v],C_2[w])$. Then, for the label in $\mathcal{T}$, one of the following four cases holds.

1) (v,w) is a label in $\mathcal{T}$.
2) (v,ε) and (v',w) are labels in $\mathcal{T}$.
3) (ε,w) and (v,w') are labels in $\mathcal{T}$.
4) (v,ε) and (ε,w) are labels in $\mathcal{T}$.

It is not necessary to consider the case 4) because the resulting alignment to delete the two nodes and then add (v,w) as the new root, which is the case 1), has a smaller cost.

For the case 1), the root of $\mathcal{T}$ is (v,w). By the forms of $C_1[v]$ and $C_2[w]$, it holds that $\tau_{\mathrm{ALN}}(C_1[v],C_2[w]) = \gamma(v,w) + \delta_{\mathrm{ALN}}(\langle L_1(v)\mid B_1(v)\rangle,\langle L_2(w)\mid B_2(w)\rangle)$, which is the recurrence (T_3).

For the case 2), the root of $\mathcal{T}$ is (v,ε). By the form of $C_1[v]$, since $|B_1(v)|\geq 2$, $B_1(v)$, not $L_1(v)$, contains the node v' corresponding to w in $C_2[w]$. Then, $\mathcal{T}$ contains a label (v'',ε) for every $v''\in L_1(v)$. Hence, it holds that $\tau_{\mathrm{ALN}}(C_1[v],C_2[w]) = \gamma(v,\varepsilon) + \tau_{\mathrm{ALN}}(B_1(v),C_2[w]) + \delta_{\mathrm{ALN}}(\langle L_1(v)\mid\emptyset\rangle,\Phi)$, which is the recurrence (T_4). The case 3) is similar to the case 2), which is the recurrence (T_5).

Next, consider the recurrences for δ_{ALN}.

Let $\mathcal{F}$ be the optimal alignment (forest) $\mathcal{A}^*(\langle L_1\mid\emptyset\rangle,\langle L_2\mid C_2[w]\rangle)$. Then, for the label in $\mathcal{F}$, one of the following two cases holds.

1) (ε,w) is a label in $\mathcal{F}$.
2) (v,w) for some $v\in L_1$ is a label in $\mathcal{F}$.

For the case 1), by deleting w from $C_2[w]$, $\langle L_2\mid C_2[w]\rangle$ is transformed to $\langle L_2\cup L_2(w)\,|\,B_2(w)\rangle$. Hence, it holds that $\delta_{\mathrm{ALN}}(\langle L_1\mid\emptyset\rangle,\langle L_2\mid C_2[w]\rangle) = \gamma(\varepsilon,w) + \delta_{\mathrm{ALN}}(\langle L_1\mid\emptyset\rangle,\langle L_2\cup L_2(w)\,|\,B_2(w)\rangle)$, which is the recurrence (F_4).

For the case 2), once (v,w) for some $v\in L_1$ becomes a label in $\mathcal{F}$, every label in $\mathcal{F}$ for every $w'\in\langle L_2(w)\,|\,B_2(w)\rangle$ is always of the form (ε,w'). Also the labels concerned with leaves except $v\in L_1$ in $\mathcal{F}$ can be computed as $\delta_{\mathrm{ALN}}(\langle L_1\backslash\{v\}\mid\emptyset\rangle,\langle L_2\mid\emptyset\rangle)$. Hence, by selecting $v\in L_1$ with the minimum cost, we obtain the recurrence (F_5).

By using the same discussion, for the case that $\mathcal{F}$ is the optimal alignment (forest) $\mathcal{A}^*(\langle L_1\,|\,C_1[v]\rangle,\langle L_2\,|\,\emptyset\rangle)$, we obtain the recurrences (F_6) and (F_7).

Let $\mathcal{F}$ be the optimal alignment (forest) $\mathcal{A}^*(\langle L_1\mid C_1[v]\rangle,\langle L_2\mid C_2[w]\rangle)$. Since $|C_1[v]|\geq 2$ and $|C_2[w]|\geq 2$, $\mathcal{F}$ contains labels for the alignment of $C_1[v]$ and $C_2[w]$ and that of L_1 and L_2. Hence, it holds that $\delta_{\mathrm{ALN}}(\langle L_1\,|\,C_1[v]\rangle,\langle L_2\,|\,C_2[w]\rangle) = \delta_{\mathrm{ALN}}(\langle L_1\,|\,\emptyset\rangle,\langle L_2\,|\,\emptyset\rangle + \tau_{\mathrm{ALN}}(C_1[v],C_2[w])$, which is the recurrence (F_8). ■

Example 4: Consider two caterpillars C_1 and C_2 in Figure 2 in Example 1 and assume the unit cost function. By applying the recurrences in Figure 6, we obtain that the alignment distance $\tau_{\mathrm{ALN}}(C_1,C_2)$ between C_1 and C_2 is 3 illustrated in Figure 7. Here, we represent a multiset as a sequence enclosed by "[" and "]" and a caterpillar as a term-like representation with "[" and "]", that is, $C_1 = a[b,b,b]$ and $C_2 = a[b,a[b,b[a,a]]]$.

$$\begin{array}{rll} & \tau_{\mathrm{ALN}}(C_1,C_2) & \\ = & \underbrace{\gamma(a,a)}_{=0}+\delta_{\mathrm{ALN}}(\langle[b,b,b]\,|\,\emptyset\rangle,\langle[b]\,|\,a[b,b[a,a]]\rangle) & (T_3) \\ = & \underbrace{\gamma(\varepsilon,a)}_{=1}+\delta_{\mathrm{ALN}}(\langle[b,b,b]\,|\,\emptyset\rangle,\langle[b,b]\,|\,b[a,a]\rangle) & (F_4) \\ = & 1+\underbrace{\gamma(b,b)}_{=0}+\delta_{\mathrm{ALN}}(\langle[b,b]\,|\,\emptyset\rangle,\langle[b,b]\,|\,\emptyset\rangle) & \\ & \quad +\delta_{\mathrm{ALN}}(\Phi,\langle[a,a]\,|\,\emptyset\}\rangle) & (F_5) \\ = & 1+\underbrace{\mu([b,b],[b,b])}_{=0}+\underbrace{\gamma(\varepsilon,a)}_{=1}+\underbrace{\gamma(\varepsilon,a)}_{=1} & (F_2),(F_3) \\ = & 3. & \end{array}$$

Fig. 7. The result of computing $\tau_{\mathrm{ALN}}(C_1,C_2)$ in Example 4.

Example 5: Consider two caterpillars $C_3 = a[a,d[b,c]]$ and $C_4 = a[c,e[b,a]]$ in Figure 2 in Example 1 and assume the unit cost function. By applying the recurrences in Figure 6, we obtain that the alignment distance $\tau_{\mathrm{ALN}}(C_3,C_4)$ between C_3 and C_4 is 3 illustrated in Figure 8.

$$\begin{aligned}
& \tau_{\mathrm{ALN}}(C_3, C_4) \\
= \; & \underbrace{\gamma(b,b)}_{0} + \delta_{\mathrm{ALN}}(\langle [a] \,|\, a[b,c] \rangle, \langle [c] \,|\, c[b,a] \rangle) && (T_3) \\
= \; & \underbrace{\gamma(a,c)}_{1} + \delta_{\mathrm{ALN}}(\langle [a] \,|\, \emptyset \rangle, \langle [c] \,|\, \emptyset \rangle) \\
& + \delta_{\mathrm{ALN}}(\langle [b,c] \,|\, \emptyset \rangle, \langle [b,a] \,|\, \emptyset \rangle) && (F_8) \\
= \; & 1 + \underbrace{\mu([a],[c])}_{=1} + \underbrace{\mu([b,c],[a,b])}_{=1} && (F_3) \\
= \; & 3.
\end{aligned}$$

Fig. 8. The result of computing $\tau_{\mathrm{ALN}}(C_3, C_4)$ in Example 5.

C. Algorithm and time complexity

Let $C_1[v]$ and $C_2[w]$ be caterpillars. Then, we denote $bb(C_1[v])$ by a sequence $v_1, \ldots, v_n$ such that $v_n = v$ and $par(v_i) = v_{i+1}$ $(1 \leq i \leq n-1)$ and $bb(C_2[w])$ by a sequence $w_1, \ldots, w_m$ such that $w_m = w$ and $par(w_j) = w_{j+1}$ $(1 \leq j \leq m{-}1)$. In this case, we denote by $bb(C_1[v]) = [v_1, \ldots, v_n]$ and $bb(C_2[w]) = [w_1, \ldots, w_m]$. Also we use the same notations of $L_1(v_i)$ and $B_1(v_i)$ for $1 \leq i \leq n$ and $L_2(w_j)$ and $B_2(w_j)$ for $1 \leq j \leq m$.

Based on the recurrences in Figure 6, Algorithm 1 illustrates the algorithm to compute the alignment distance $\tau_{\mathrm{ALN}}(C_1, C_2)$ between caterpillars C_1 and C_2. Here, the statement "$v \leftarrow$ (A)" means to substitute the value of computing the right side of the recurrence (A) to v, for example.

```
procedure $\tau_{\mathrm{ALN}}(C_1, C_2)$
    /* $C_1, C_2$: caterpillars, $bb(C_1) = [v_1, \ldots, v_n]$,
    $bb(C_2) = [w_1, \ldots, w_m]$, $v_n = r(C_1)$, $w_m = r(C_2)$ */
    $\tau_{\mathrm{ALN}}(\emptyset, \emptyset) \leftarrow 0$; /* $(T_0)$ */
    for $i = 1$ to $n$ do  $\tau_{\mathrm{ALN}}(C_1[v_i], \emptyset) \leftarrow (T_1)$;
    for $j = 1$ to $m$ do  $\tau_{\mathrm{ALN}}(\emptyset, C_2[w_j]) \leftarrow (T_2)$;
    for $i = 1$ to $n$ do
        for $j = 1$ to $m$ do
            $\tau_{\mathrm{ALN}}(C_1[v_i], C_2[w_j]) \leftarrow (A)$;

procedure $\delta_{\mathrm{ALN}}(\langle L_1 \,|\, C_1 \rangle, \langle L_2 \,|\, C_2 \rangle)$
    /* $L_1, L_2$ : set of leaves, $C_1, C_2$: caterpillars */
    if $C_1 = \emptyset$ and $C_2 = \emptyset$ then
        $\delta_{\mathrm{TAI}}(\langle L_1 \,|\, \emptyset \rangle, \langle L_2 \,|\, \emptyset \rangle) \leftarrow (F_3)$;
    else if $C_1 \neq \emptyset$ and $C_2 = \emptyset$ then
        /* $bb(C_1) = [v_1, \ldots, v_n]$, $v_n = r(C_1)$ */
        if $L_2 = \emptyset$ then  $\delta_{\mathrm{TAI}}(\langle L_1 \,|\, C_1 \rangle, \Phi) \leftarrow (F_1)$;
        else
            for $i = 1$ to $n$ do
                $\delta_{\mathrm{TAI}}(\langle L_1 \,|\, C_1[v_i] \rangle, \langle L_2 \,|\, \emptyset \rangle) \leftarrow (B)$;
    else if $C_1 = \emptyset$ and $C_2 \neq \emptyset$ then
        /* $bb(C_2) = [w_1, \ldots, w_m]$, $w_m = r(C_2)$ */
        if $L_1 = \emptyset$ then  $\delta_{\mathrm{TAI}}(\Phi, \langle L_2 \,|\, C_2 \rangle) \leftarrow (F_2)$;
        else
            for $j = 1$ to $m$ do
                $\delta_{\mathrm{TAI}}(\langle L_1 \,|\, \emptyset \rangle, \langle L_2 \,|\, C_2[w_j] \rangle) \leftarrow (C)$;
    else
        /* $bb(C_1) = [v_1, \ldots, v_n]$, $bb(C_2) = [w_1, \ldots, w_m]$,
        $v_n = r(C_1)$, $w_m = r(C_2)$ */
        for $i = 1$ to $n$ do
            for $j = 1$ to $m$ do
                $\delta_{\mathrm{TAI}}(\langle L_1 \,|\, C_1[v_i] \rangle, \langle L_2 \,|\, C_2[w_j] \rangle) \leftarrow (D)$;
```

Algorithm 1: $\tau_{\mathrm{ALN}}(C_1, C_2)$

Theorem 4: Let C_1 and C_2 be caterpillars, where $h = \max\{h(C_1), h(C_2)\}$ and $\lambda = \max\{|lv(C_1)|, |lv(C_2)|\}$. Then, we can compute the alignment distance $\tau_{\mathrm{ALN}}(C_1, C_2)$ between C_1 and C_2 in $O(h^2\lambda^3)$ time. Furthermore, if we adopt the unit cost function, then we can compute it in $O(h^2\lambda)$ time.

Proof: Let $bb(C_1) = [v_1, \ldots, v_n]$ and $bb(C_2) = [w_1, \ldots, w_m]$. Then, it is obvious that $h(C_1) = n + 1$ and $h(C_2) = m + 1$, so it holds that $m \leq h - 1$ and $n \leq h - 1$.

The algorithm of computing $\tau_{\mathrm{ALN}}(C_1, C_2)$ calls every pair $(v_i, w_j) \in bb(C_1) \times bb(C_2)$ just once. When computing $\delta_{\mathrm{ALN}}(\langle L_1(v_{i-1}) \,|\, C_1[v_i] \rangle, \langle L_2(w_{j-1}) \,|\, C_2[w_j] \rangle)$ for $2 \leq i \leq n$ and $2 \leq j \leq m$, it is possible to construct multisets $S_1 = L_1(v_1) \sqcup \cdots \sqcup L_1(v_{i-1})$ and $S_2 = L_2(w_1) \sqcup \cdots \sqcup L_2(w_{j-1})$ and compute the edit distance $\mu(S_1, S_2)$ between multisets in the worst case. By Lemma 2, we can compute it in $O(\lambda^3)$ time under the general cost function and in $O(\lambda)$ time under the unit cost function.

Hence, the total running time of computing $\tau_{\mathrm{ALN}}(C_1, C_2)$ under the general cost function is described as follows:

$$\sum_{i=1}^{n} \sum_{j=1}^{m} O(\lambda^3) = O(\lambda^3)mn \leq O(\lambda^3)(h-1)^2 = O(h^2\lambda^3).$$

By replacing $O(\lambda^3)$ with $O(\lambda)$, this time complexity is reduced to $O(h^2\lambda)$ time under the unit cost function. ■

Theorem 4 also claims that the structural restriction of caterpillars provides the limitation of tractable computing the alignment distance for unordered trees as follows. We say that a tree is a *generalized caterpillar* if it is transformed to a caterpillar after removing all the leaves in it. Then, the following theorem also holds as a corollary of [8].

Theorem 5 (cf., [1], [4]): The problems of computing the alignment distance τ_{ALN} between generalized caterpillars are MAX SNP-hard, even if the maximum height is at most 3 and the cost function is the unit cost function.

Proof: It is straightforward from the proof of Corollary 4.3 in [1] or Theorem 1 in [4] and because the Tai mapping constructed in their proof is a less-constrained mapping. ■

V. Conclusion and Future Works

In this paper, we have designed the algorithm to compute the alignment distance τ_{ALN} between caterpillars in $O(h^2\lambda^3)$ time under the general cost function and in $O(h^2\lambda)$ time under the unit cost function.

It is an important future work to implement the algorithms and then give experimental results to compute τ_{ALN}, with comparing the results of τ_{TAI} in [8] with those of τ_{ALN}. Since the proof in Theorem 4 is rough, it is possible to improve the time complexity, together with that of computing the edit distance between multisets under the general cost function (Lemma 2), which is also a future work.

References

[1] T. Akutsu, D. Fukagawa, M. M. Halldórsson, A. Takasu, K. Tanaka: *Approximation and parameterized algorithms for common subtrees and edit distance between unordered trees*, Theoret. Comput. Sci. **470**, 10–22 (2013). https://doi.org/10.1016/j.tcs.2012.11.017.

[2] M. M. Deza, E. Deza: *Encyclopedia of distances* (4th ed.) Springer (2016). https://doi.org/10.1007/978-3-662-52844-0.

[3] J. A. Gallian: *A dynamic survey of graph labeling*, Electorn. J. Combin., DS6 (2018).

[4] K. Hirata, Y. Yamamoto, T. Kuboyama: *Improved MAX SNP-hard results for finding an edit distance between unordered trees*, Proc. CPM'11, LNCS **6661**, 402–415 (2011). https://doi.org/10.1007/978-3-642-21458-5_34.

[5] T. Jiang, L. Wang, K. Zhang: *Alignment of trees – an alternative to tree edit*, Theoret. Comput. Sci. **143**, 137–148 (1995). https://doi.org/10.1016/0304-3975(95)80029-9.

[6] T. Kuboyama: *Matching and learning in trees*, Ph.D thesis, University of Tokyo (2007).

[7] C. L. Lu, Z.-Y. Su, C. Y. Yang: *A new measure of edit distance between labeled trees*, Proc. COCOON'01, LNCS **2108**, 338–348 (2001). https://doi.org/10.1007/3-540-44679-6_37.

[8] K. Muraka, T. Yoshino, K. Hirata: *Computing edit distance between rooted labeled caterpillars*, Proc. FedCSIS'18, 245–252 (2018). http://dx.doi.org/10.15439/2018F179.

[9] K. Muraka, T. Yoshino, K. Hirata: *Vertical and horizontal distance to approximate edit distance for rooted labeled caterpillars*, Proc. ICPRAM'19, 590–597 (2019). https://dx.doi.org/10.5220/0007387205900597.

[10] K.-C. Tai: *The tree-to-tree correction problem*, J. ACM **26**, 422–433 (1979). https://doi.org/10.1145/322139.322143.

[11] J. T. L. Wang, K. Zhang: *Finding similar consensus between trees: An algorithm and a distance hierarchy*, Pattern Recog. **34**, 127–137 (2001). https://doi.org/10.1016/50031-3203(99)00199-5.

[12] Y. Yamamoto, K. Hirata, T. Kuboyama: *Tractable and intractable variations of unordered tree edit distance*, Internat. J. Found. Comput. Sci. **25**, 307–330 (2014). https://doi.org/10.1142/50129054114500154.

[13] T. Yoshino, K. Hirata: *Tai mapping hierarchy for rooted labeled trees through common subforest*, Theory Comput. Sys. **60**, 759–783 (2017). https://doi.org/10.1007/s00224-016-9705-1.

[14] K. Zhang: *A constrained edit distance between unordered labeled trees*, Algorithmica **15**, 205–222 (1996). https://doi.org/10.1007/BF01975866.

[15] K. Zhang, T. Jiang: *Some MAX SNP-hard results concerning unordered labeled trees*, Inform. Process. Lett. **49**, 249–254 (1994). https://doi.org/10.1016/0020-0190(94)90062-0.

[16] K. Zhang, J. Wang, D. Shasha: *On the editing distance between undirected acyclic graphs*, Internat. J. Found. Comput. Sci. **7**, 45–58 (1996). https://doi.org/10.1142/S0129054196000051.

Appendix

In this appendix, we point out the number of caterpillars in real data and revise the result for the edit distance between caterpillars in [8].

A. Caterpillars in real data

Table I, which is represented in [8], illustrates the number of caterpillars in N-glycans and all glycans from KEGG[2], CSLOGS[3], dblp[4], and SwissProt, TPC-H, Auction, Nasa, Protein and University from UW XML Repository[5]. Here, #cat is the number of caterpillars and #data is the total number of data. For $D \in \{\text{Auction, Nasa, Protein, University}\}$, D^- denotes the trees obtained by deleting the root for every tree in D. Since one tree in D produces some trees in D^-, the total number of trees in D^- is greater than that of D. Hence, there are some cases containing many caterpillars in real dataset.

TABLE I
THE NUMBER OF CATERPILLARS IN N-GLYCANS AND ALL GLYCANS FROM KEGG, CSLOGS, DBLP, SWISSPROT, TPC-H, AUCTION, UNIVERSITY, PROTEIN AND NASA.

dataset	#cat	#data	%
N-glycans	514	2,142	23.996
all glycans	8,005	10,704	74.785
CSLOGS	41,592	59,691	69.679
dblp	5,154,295	5,154,530	99.995
SwissProt	6,804	50,000	13.608
TPC-H	86,805	86,805	100.000
Auction	0	37	0
Nasa	0	2,430	0
Protein	0	262,625	0
University	0	6,738	0
Auction$^-$	259	259	100.000
Nasa$^-$	21,245	27,921	76.089
Protein$^-$	1,874,703	2,204,068	85.057
University$^-$	74,638	79,213	94.224

B. The revision of the edit distance for caterpillars

Muraka *et al.* [8] have designed the algorithm to compute the edit distance $\tau_{\mathrm{TAI}}(C_1, C_2)$. Then, they have pointed out that its time complexity is $O(h^2\lambda^2)$ time, where $h = \max\{h(C_1), h(C_2)\}$ and $\lambda = \max\{|lv(C_1)|, |lv(C_2)|\}$.

Note that their recurrence between the set of leaves is based on the string edit distance. However, as similar as the alignment distance in this paper, in order to compute the edit distance between the set of leaves, it is necessary to adopt the edit distance for multisets.

Let $s(L)$ be the string representation of the set L of leaves and σ the string edit distance. Then, Muraka *et al.* [8] have introduced the following recurrence to compute $\tau_{\mathrm{TAI}}(C_1, C_2)$:

$$\delta_{\mathrm{TAI}}(\langle L_1 \,|\, \emptyset\rangle, \langle L_2 \,|\, \emptyset\rangle) = \sigma(s(L_1), s(L_2)).$$

On the other hand, as stated above, it is necessary to replace this recurrence with the following recurrence to compute $\tau_{\mathrm{TAI}}(C_1, C_2)$ as same as Figure 6:

$$\delta_{\mathrm{TAI}}(\langle L_1 \,|\, \emptyset\rangle, \langle L_2 \,|\, \emptyset\rangle) = \mu(\widetilde{L_1}, \widetilde{L_2}).$$

Consider the time complexity of $O(h^2\lambda^2)$ presented in [8]. The part $O(\lambda^2)$ follows from the time complexity of computing the string edit distance between the set of (all the) leaves in two caterpillars. On the other hand, by replacing the recurrence as above, it is necessary to revise this part based on the time complexity of computing the multiset edit distance, that is, revise to $O(\lambda^3)$ under the general cost function and $O(\lambda)$ under the unit cost function by Lemma 2. Furthermore, we can improve the proof of [8] to the similar proof of Theorem 4.

Hence, we can compute $\tau_{\mathrm{TAI}}(C_1, C_2)$ in $O(h^2\lambda^3)$ time under the general cost function and $O(h^2\lambda)$ time under the unit cost function, which is the same result of $\tau_{\mathrm{ALN}}(C_1, C_2)$.

[2] Kyoto Encyclopedia of Genes and Genomes, http://www.kegg.jp/

[3] http://www.cs.rpi.edu/~zaki/www-new/pmwiki.php/Software/Software

[4] http://dblp.uni-trier.de/

[5] http://aiweb.cs.washington.edu/research/projects/xmltk/xmldata
/www/repository.html

Advances in Computer Science & Systems

CSS is a FedCSIS conference track aiming at integrating and creating synergy between FedCSIS events that thematically subscribe to more technical aspects of computer science and related disciplines. The CSS track spans themes ranging from hardware issues close to the discipline of computer engineering via software issues tackled by the theory and applications of computer science and to communications issues of interest to distributed and network systems. Technical sessions that constitute CSS are:

- CANA'19—12th Workshop on Computer Aspects of Numerical Algorithms
- C&SS'19—6th International Conference on Cryptography and Security Systems
- LTA'19—4th International Workshop on Language Technologies and Applications
- MMAP'19—12th International Symposium on Multimedia Applications and Processing
- WAPL'19 7th Workshop on Advances in Programming Languages
- WSC'19—10th Workshop on Scalable Computing

12th Workshop on Computer Aspects of Numerical Algorithms

NUMERICAL algorithms are widely used by scientists engaged in various areas. There is a special need of highly efficient and easy-to-use scalable tools for solving large scale problems. The workshop is devoted to numerical algorithms with the particular attention to the latest scientific trends in this area and to problems related to implementation of libraries of efficient numerical algorithms. The goal of the workshop is meeting of researchers from various institutes and exchanging of their experience, and integrations of scientific centers.

TOPICS

- Parallel numerical algorithms
- Novel data formats for dense and sparse matrices
- Libraries for numerical computations
- Numerical algorithms testing and benchmarking
- Analysis of rounding errors of numerical algorithms
- Languages, tools and environments for programming numerical algorithms
- Numerical algorithms on coprocesors (GPU, Intel Xeon Phi, etc.)
- Paradigms of programming numerical algorithms
- Contemporary computer architectures
- Heterogeneous numerical algorithms
- Applications of numerical algorithms in science and technology

EVENT CHAIRS

- **Bylina, Beata,** Maria Curie-Sklodowska University, Poland
- **Bylina, Jaroslaw,** Maria Curie-Sklodowska University, Poland
- **Stpiczyński, Przemysław,** Maria Curie-Sklodowska University, Poland

PROGRAM COMMITTEE

- **Amodio, Pierluigi,** Università di Bari, Italy
- **Anastassi, Zacharias,** De Montfort University, United Kingdom
- **Brugnano, Luigi,** Universita' di Firenze, Italy
- **Fialko, Sergiy,** Tadeusz Kościuszko Cracow University of Technology, Poland
- **Georgiev, Krassimir,** IICT - BAS, Bulgaria
- **Gravvanis, George,** Democritus University of Thrace, Greece
- **Kozielski, Stanislaw,** Silesian University of Technology, Poland
- **Lirkov, Ivan,** Institute of Information and Communication Technologies, Bulgarian Academy of Sciences, Bulgaria
- **Luszczek, Piotr,** University of Tennessee, United States
- **Marowka, Ami,** Bar-Ilan University, Israel
- **Petcu, Dana,** West University of Timisoara, Romania
- **Sergeichuk, Vladimir,** Institute of Mathematics of NAS of Ukraine, Ukraine
- **Srinivasan, Natesan,** Indian Institute of Technology, India
- **Tudruj, Marek,** Inst. of Comp. Science Polish Academy of Sciences/Polish-Japanese Institute of Information Technology, Poland
- **Tůma, Miroslav,** Academy of Sciences of the Czech Republic, Czech Republic
- **Vazhenin, Alexander,** University of Aizu, Japan

DOI: 10.15439/2019F344
ISSN 2300-5963 ACSIS, Vol. 19

Performance and Energy Evaluation of Parallel Particle Simulation Algorithms for Different Input Particle Data

Robert Kiesel, Gudula Rünger
Department of Computer Science
Chemnitz University of Technology
Chemnitz, Germany
Email: {robert.kiesel, ruenger}@informatik.tu-chemnitz.de

***Abstract*—Particle simulations are popular methods for the simulation of applications from a wide range of sciences, including astrophysics, biology or chemistry. Usually, these applications require a large number of simulation steps, each of which computes a change of the entire particle system. Depending on the number of simulation steps and also the size and structure of the specific particle system, the computation time can be quite large and the exploitation of parallel architectures is usually necessary. In this article, we investigate the performance and energy consumption for different particle simulation methods and distinguish different input particle data. The investigations are done for the particle simulation methods from the ScaFaCoS library and use the various input data of homogeneous or inhomogeneous nature. Experiments are performed on multicore systems.**

I. Introduction

PARTICLE SIMULATIONS are popular methods for simulating various scientific problems from areas, such as astrophysics, biology or chemistry. The computation time for particle simulations can be quite large, especially for simulating long-range particle interactions, for example occurring in gravitational or Coulomb interactions, since the direct computation for a simulation with N particles has the complexity $\mathcal{O}(N^2)$. Efficient implementations of the particle simulation often split the computation with respect to a cut-off radius, which means that only the particle interactions of particles lying within the cut-off radius are computed exactly and the interactions of particles with a distance larger than the cut-off radius are computed by an approximation. The computation time can be reduced to $\mathcal{O}(N \log N)$ or $\mathcal{O}(N)$ with efficient methods, such as the Fast Multipole Method (FMM) [1] or the fast Fourier-transform (FFT) [2].

However, the actual execution time of a particle simulation depends on the given size N of the particle system and the structure of the initial distribution of the N particles in the particle system. Thus, for a fast simulation the simulation method has to be chosen carefully to be suitable for the size and characteristics of the particle input set. For very small particle systems, the direct computation could still be the fastest, since there is no splitting overhead with respect to the cut-off radius. Also, a different simulation method might be suitable for particle systems of different size N which have the same characteristics with respect to the particle distribution. Concerning the initial distribution, the particle systems are often distinguished being homogeneous, inhomogeneous or a mix of both with parts of the particle system being homogeneous but also containing some inhomogeneous regions. While in homogeneous particle systems, the particles are equally distributed in the entire particle system, an inhomogeneous particle system can exhibit distributions in which the particles are clustered in certain areas. Since particle simulations are time-step based algorithms which compute a new particle situation in each step, the structure of a particle system can change during the simulation process. Thus, different simulation methods might be suitable at different points in simulation time so that exchanging the method after some time steps could be beneficial. To support the adaptation of the simulation method, it is required to know in which situation which setting might lead to the desired performance improvement.

Naturally, the hardware platform has a large influence on the performance and accelerators, such as GPUs can be exploited when implementation variants for GPUs, e.g., with CUDA, OpenCL or OpenACC, are available. However, using GPUs requires a transfer of data to the device which might cause a big overhead, and thus is not always advantageous. Depending on the size and structure of the input particle data and availability of hardware, e.g., a CPU or a GPU, the usage of a specific method on specific hardware has to be chosen. For the grid-based methods there exists an OpenCL implementation to use GPUs for the near-field, but these methods are, in consequence of the regular grid over the complete particle system, designed for homogeneous systems. In contrast, tree-based methods are not dependent on a homogeneous system.

In this article, we consider different particle simulation methods of the Scalable Fast Coulomb Solvers (ScaFaCoS) library [3] and study their performance and energy consumption for various particle systems. The ScaFaCoS library contains parallel implementations of efficient solver methods for long-range particle interactions. The parallelizations use the Message Passing Interface (MPI). Additional parallel implementations are available for selected modules. An example is the OpenCL implementation of the near-field module for grid-based methods [4], which allows an execution on various hardware platforms, such as GPUs.

The objective of our work is to investigate the performance and energy consumption behavior of selected particle simulation methods with respect to the characteristics of the input particle system. The contribution of this article includes the detailed measurements and investigation of different algorithms for particle simulation for different input sets and hybrid hardware platforms. The work is meant to provide a rich data basis of performance and energy data for future tuning approaches.

The rest of this article is organized as follows: Section II introduces the particle methods used. Section III describes the generation of the particle systems used. Section IV presents the experimental results. Section V summarizes the performance results. Section VI discusses related work. Section VII concludes the article and discusses the tuning potential.

II. Particle simulation methods

The particle simulation method is a general solution method to simulate all kinds of problems which can be represented by a set of so-called particles which react to each other according to problem-specific rules. For several decades, different versions of particle solution methods have been invented and different implementations of these methods have been developed, all of which solve particle problems but may have a different performance. In this article, we concentrate on particle simulation methods being implemented in the ScaFaCoS library.

A. Particle models

Particle models are simulation models in which physical phenomena are described by a discrete representation of interacting particles. A particle has usually problem specific attributes, such as position, mass, momentum or velocity. The motion in the physical system is calculated in a series of simulation time steps each of which computes one interaction event between the particles by recomputing the attribute values. Such particle models have been used to explain properties of solids, liquids or gases represented by a finite input set of particles and corresponding rules for the specific interaction. Usually, the number of particles is constant for one simulation run, and thus there is no need for updates during one specific simulation. Three principal types of particle simulation models have been identified, which are the particle-particle model using action at a distance, the particle-mesh model using an approximation by a mesh and the $\mathrm{P}^3\mathrm{M}$ model being a combination of both. The specific use of the simulation type depends on the physical model to be simulated and also on the computational cost for a computer simulation.

In this article, we consider molecular dynamics simulations for Coulomb forces which are characterized by long-range interactions. In the simulation of long-range interactions, the number of interactions per simulation time step is not limited to particles in the proximity. Thus, all pairwise interactions between all particles in the system have to be calculated in each time step, which leads to a computationally expensive simulation. This problem can be treated by hierarchical approximation algorithms reducing the quadratic complexity to a linear complexity or by parallel implementations on different parallel devices. In our investigations, we use MPI implementations but also parallel hybrid implementations on CPU/GPU for particle solvers from the ScaFaCoS library.

B. The ScaFaCoS library

The Scalable Fast Coulomb Solvers (ScaFaCoS) library contains parallel implementations for several different particle simulation methods, e.g., a Direct method, Particle-Particle Particle-Mesh ($\mathrm{P}^3\mathrm{M}$), Particle-Particle nonequispaced fast Fourier transforms ($\mathrm{P}^2\mathrm{NFFT}$), Fast Multipole Method (FMM) or Pretty Efficient Parallel Coulomb Solver (PEPC). The ScaFaCoS library is fully parallelized using MPI and with certain parts using OpenCL.

The direct computation, e.g., a pairwise interaction between all N particles, requires $O(N^2)$ operations. More efficient methods are reducing this complexity by using approximation approaches which split the calculation into a near-field and a far-field part with respect to each particle. The implementation of this splitting into near- and far-field are different for different particle solution methods. While in the near-field part the pairwise interaction between particles is computed, the far-field part might be computed approximately leading to a more efficient computation. A comparison of the solver methods of the ScaFaCoS library is given in [5].

Besides the accuracy of the computations and also some solver specific parameters, the performance of particle simulations depends on the particle distribution of the particle system. If the particles are equally distributed in the particle system, it is called a homogeneous particle system and usually less expensive to simulate for the particle solvers, in contrast to inhomogeneous particle systems, in which particles are irregularly distributed in the particle system. The input data for the solvers in the ScaFaCoS library describe the corresponding particle system by attributes consisting of the charge value and the three-dimensional position of each particle. In this article, the emphasis is on three of the ScaFaCoS simulation methods, which are the direct method, which is a particle to particle calculation, the $\mathrm{P}^2\mathrm{NFFT}$, which is a fourier based approach, and the FMM method, which is tree based to reduce the complexity.

C. Fourier based

Fourier-based methods compute the far-field computations in Fourier space, mostly by using the fast Fourier-transforms (FFT). The method $\mathrm{P}^2\mathrm{NFFT}$ [2] is used as an example method for the Fourier-based approach. The computational demands of the far-field and the near-field parts are influenced by parameters that specify the size of the FFT grid and the cut-off range. Far-field potentials are computed via convolution in Fourier space. The computation of the near-field interactions is calculated by the ScaFaCoS near-field module, which computes pairwise interactions. The P^{2}NFFT and P^{3}M implementation of the ScaFaCoS library are using the same near-field implementation. An OpenCL implementation for this near-field has been developed to use both Multicore-CPUs and GPUs in [4].

D. Tree based

Another approach to split the particle system is possible by using an octree structure. The particles are sorted into spatial boxes respective to their position in the particle system. The boxes are then organized into an octree which is exploited to compute the interactions on different levels. Since this approach does not split the particle system into a regular box system, it works on homogeneous systems as well as on inhomogeneous input particle systems. The FMM [1] is an example for this approach and has been implemented in ScaFaCoS. This FMM version has its own near-field implementation for which for which an OpenCL version for execution on GPU does not exists.

For a specific particle, the near field potential is determined by calculating the potential at the position of the particle caused by each of the particles in the same and neighboring octree boxes. The far-field potential is calculated using approximate values of the potential caused by all particles in a particular octree box. These approximations are calculated for each octree level. The approximations at appropriate octree levels are then used to approximate the far-field potential at a particular particle position. The tree depth determines the separation in the near-field and the far-field potential and, thus, the tree depth is an important parameter for the accuracy as well as for the performance of a simulation run.

III. Generation of particle systems

The particle systems used have particles that are Hammersley distributed [6] to ensure a minimal space between the particles. All tested particle systems are periodical, i.e., if a particle leaves the particle system, a new particle enters the system on the opposite side.

The Hammersley distributed particle systems are generated with the formulas given in this section as described in [7].

If p is a prime number, each nonnegative number k can be displayed as:

$$k = a_0 + a_1 p + a_2 p^2 + \cdots + a_r p^r \tag{1}$$

with $a_i \in \{0, \ldots, p-1\}, \quad i = 0, \ldots, r, \quad r \in \mathbb{N}$.

A function $\phi(k)$ can be defined as follows:

$$\phi_p(k) = \frac{a_0}{p} + \frac{a_1}{p^2} + \frac{a_2}{p^3} + \cdots + \frac{a_r}{p^{r+1}} \tag{2}$$

The following explains the Hammersley distribution:

We define d as the dimension of the data to be generated and $p_1, p_2, \ldots, p_{d-1}$ the prime numbers with $p_1 < p_2 < \cdots < p_{d-1}$. N is the number of particles to be generated. The particle k is defined as follows:

$$\left(\frac{k}{N}, \phi_{p_1}(k), \ldots, \phi_{p_{d-1}}(k)\right), k = 0, \ldots, N-1 \tag{3}$$

Since the first component of particle k depends on N , the number of particles has to be set before the generation starts.

The particle systems used are distributed in a cube of size $[0,1]^3$ with different Hammersley distributions.

There exists an implementation named HAMMERSLEY[1] which can provide different Hammersley distributions, e.g., Ball, Two Balls, Grid Face and Cube as shown in Figure 1. The four different distributions used in this article are generated as follows:

- **Cube:** The particles are Hammersley distributed in the whole particle cube.
- **Grid Face:** The number of particles is $N = 4 \cdot N_c^3, N_c \in \mathbb{N}$ and j is defined as:
$$\begin{aligned} j &= (N_c \cdot u + v) \cdot N_c + w \\ &(u, v, w \in \{0, \ldots, N_c - 1\}) \end{aligned} \tag{4}$$
The positions of the particles x_{4j+1} to x_{4j+4} are then defined as follows:
$$\begin{aligned} x_{4j+1} &= (u, v, w)^T / (N_c - 0.5) \\ x_{4j+2} &= (u + 0.5, v + 0.5, w)^T / N_c - 0.5 \\ x_{4j+3} &= (u + 0.5, v, w + 0.5)^T / N_c - 0.5 \\ x_{4j+4} &= (u, v + 0.5, w + 0.5)^T / N_c - 0.5 \end{aligned} \tag{5}$$
- **Ball:** Inside of the particle cube, the particles are Hammersley distributed in the following ball:
$$(x - 0.5)^2 + (y - 0.5)^2 + (z - 0.5)^2 <= (0.5)^2 \tag{6}$$
- **Two Balls:** Two Balls of different sizes are created as balls as in Formula 6. The first Ball with $(1 - 1/64) \cdot N$ particles and the second with $1/64 \cdot N$ particles. They get a distance of 20 and are acurately scaled and shifted in the particle cube.

The charge $q_i \in \{-1; 1\}$ of each particle i is generated randomly, such that the following holds:

$$\sum_{i=1}^{N} q_i \in \{-1; 0; 1\} \tag{7}$$

To use the generated particle systems with the ScaFaCoS test program we used, they have to be converted into XML files as input data, containing the position and the actual charge of each particle.

IV. Performance results

The performance of different solvers is tested for various particle systems, i.e., two homogeneous and two inhomogeneous systems. For the Fourier-based solver also the OpenCL variant is tested. The experiments are split by the particle system distributions. To look at the solvers in more detail, the Fourier-based algorithms are executed with different solver specific parameter settings.

[1] HAMMERSLEY. The Hammersley Quasirandom Sequence. people.scs.fsu.edu/~burkardt/cpp_src/hammersley/hammersley.html

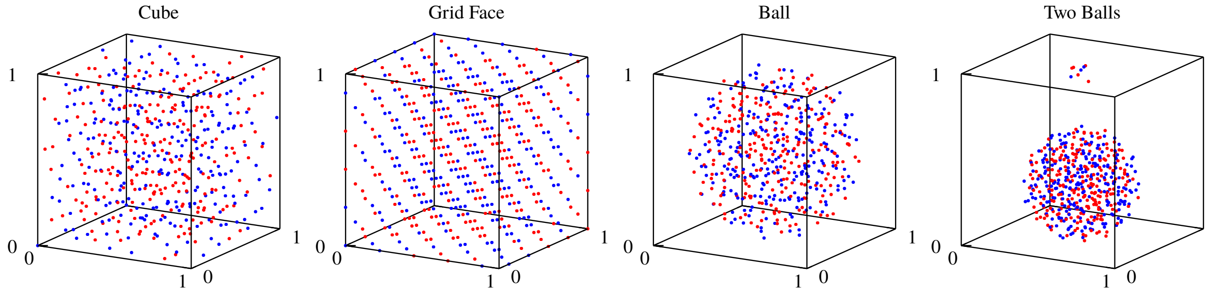


Figure 1. Illustration of two homogeneous particle distributions (left) and two inhomogeneous particle distributions (right) with positive and negative charges.

A. Experimental setup

The experiments are performed on a multicore system with four GPUs. The Haswell system consists of two Intel Xeon E5-2683 v3 processors with 14 cores each, which have 2.0 GHz. The system is equipped with four Nvidia GeForce GTX Titan Blacks. The energy consumption is measured using PAPI 5.6.0 and the RAPL interface to read the appropriate MSR registers. The energy measurements do only include the CPU, i.e., it does not include the energy consumption of the DRAM or any other component in the system. The energy consumption of the GPUs is not measured. All measurements are repeated 5 times to obtain the shown average values. For the measurements, the number of MPI processes is set to 56, which is equal to the number of cores on the Haswell system plus Hyperthreading. The frequency is set to 2.0 GHz and Intel Turbo Boost is disabled for the experiments.

B. Homogeneous systems

In homogeneous particle systems, the particles are uniformly distributed in the particle system without irregularities. The particles are not grouped into multiple clusters. Figure 1 (left) shows the two homogeneous particle systems Cube and Grid Face. The positive and negative charges are randomly generated.

Figure 2 shows the runtime and energy consumption for the FMM and P^{2}NFFT (MPI and OpenCL variant) solver with the two homogeneous systems with varying number of particles. For small particle systems, i.e., less than 50,000 particles, the P^{2}NFFT solver has a low runtime. If the particle system has more particles, the FMM solver outperforms the P^{2}NFFT. Since the transfer to the GPU takes some time, it is only useful to use the GPU with big particle systems. However for the big systems, the FMM algorithm outperforms the P^{2}NFFT even when GPUs are used. If the CPU would be less powerful, the OpenCL variant of the P^{2}NFFT should outperform the MPI variant of the FMM with bigger particle systems. The energy consumption shows the same behaviour as the runtime for homogeneous systems.

Figure 3 shows parameter tests for the P^{2}NFFT solver (MPI variant) for the two homogeneous particle systems with different system sizes. The grid size is varied from 128 to 512 to determine the best grid size for each system. The more particles a system has, the bigger the best grid size. Since there is only one global minimum and no other local minimum, simple tuning algorithms can be used to find that minimum. This minimum can be different for the runtime and the energy consumption. As the 50,000 particle system shows, the explicit best grid size can differ with the particle system structure size, e.g., 512 for the Grid Face but 448 for the Cube.

C. Inhomogeneous systems

In inhomogeneous particle systems, particles are irregularly distributed in the particle system. For example, they are clustered in single or multiple regions, thus, there are also empty regions in the field. Figure 1 (right) shows two inhomogeneous particle systems. The left particle system is a single big ball in the centre of the system, while the right particle systems is a dense ball and a smaller additional ball with less particles in distance. The right system has a bigger empty region in the particle system than the left system.

Figure 4 tests the two inhomogeneous particle systems with varying particle system sizes. As expected, the P^{2}NFFT algorithm has more problems, in terms of runtime and energy consumption, with inhomogeneous systems compared to homogeneous systems. The Ball particle system has a similar behaviour like the homogeneous systems, but with the Two Balls system P^{2}NFFT has a worse runtime and energy consumption compared to FMM, even with few particles. The GPU variant of the P^{2}NFFT has a better runtime for big particle systems than the MPI variant but is still slower than the FMM method using MPI. Like with homogeneous systems, the energy consumption shows the same behaviour. Consequently, the Ball particle system is homogeneous enough for the P^{2}NFFT algorithm, but for more inhomogeneous systems, like the Two Balls system, the FMM algorithm is better.

Figure 5 shows parameter tests for the P^{2}NFFT system for the two inhomogeneous particle systems with different system sizes. The grid size is varied from 128 to 512 to determine the best grid size for the particle system. The more particles a system has, the bigger the best grid size. The Two Balls system with 5,000 particles shows that the runtime and energy consumption can have different settings, i.e., grid size of 384 for the shortest runtime, but 448 for the lowest energy consumption.

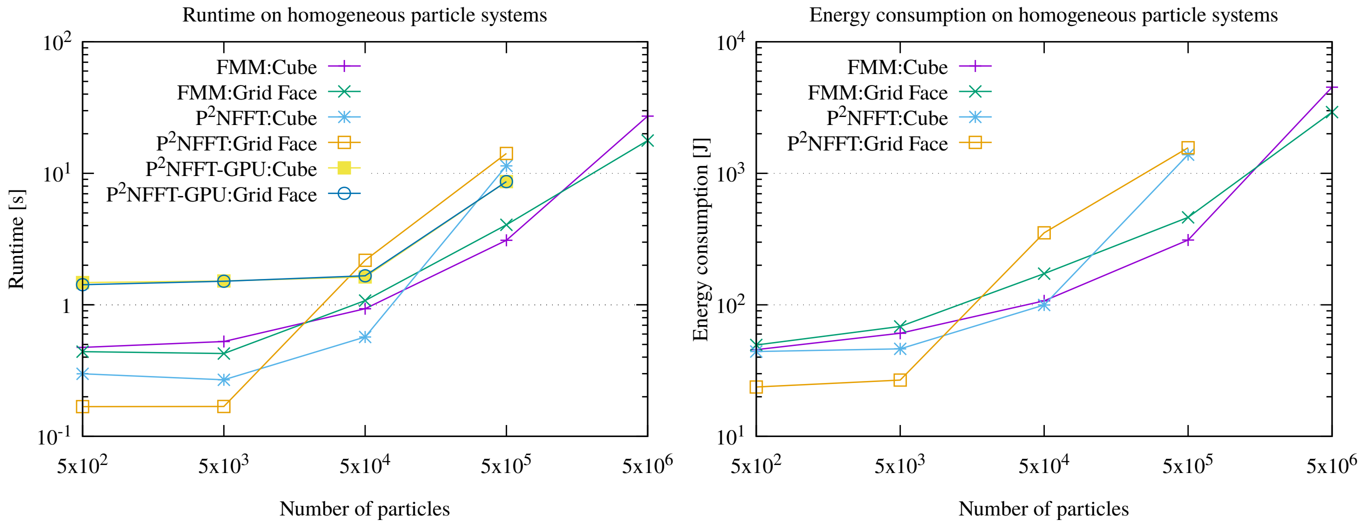


Figure 2. Parallel runtime (left) and energy measurements (right) for homogeneous systems with 56 MPI processes on the Haswell system and the Geforce GTX Titan Black.

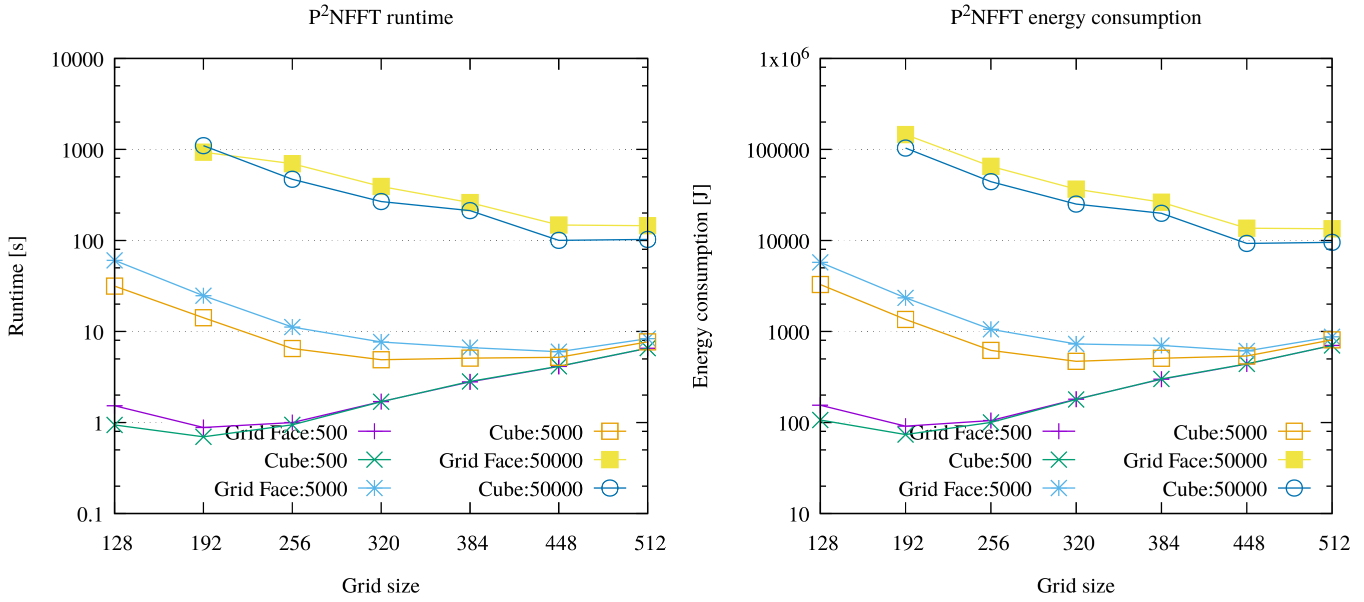


Figure 3. Parallel runtime (left) and energy measurements (right) for homogeneous systems with varying grid size with 56 MPI processes on the Haswell system.

V. Summary of the performance results

The measurement results from Section IV have shown that the performance of the different particle simulation implementations strongly depend on the execution platform as well as on the characteristics of the input particle system. Depending on the optimizing goal, the availability of the hardware and the prior knowledge of the particle system distribution and size, some decisions can be made to achieve the best performance or lowest energy consumption. Thus, based on the measurements an appropriate particle simulation method can be selected. The following observation show how it can be decided whether the FMM, the P^2NFFT or the P^2NFFT with GPU solver should be used for best performance. Selection strategies might help to select the simulation algorithm with the best execution time and/or energy consumption.

Table I
Solver selection

less than 50,000 particles	
homogeneous distribution	inhomogeneous distribution
P^2NFFT solver	FMM solver
more than 50,000 particles	
strong CPU	weak CPU but GPU available
FMM solver	P^2NFFT solver on GPU

Table I summarizes the selection of a particle simulation solver for the given HPC system. If the particle system has less than 50,000 particles, the selection of the best performing

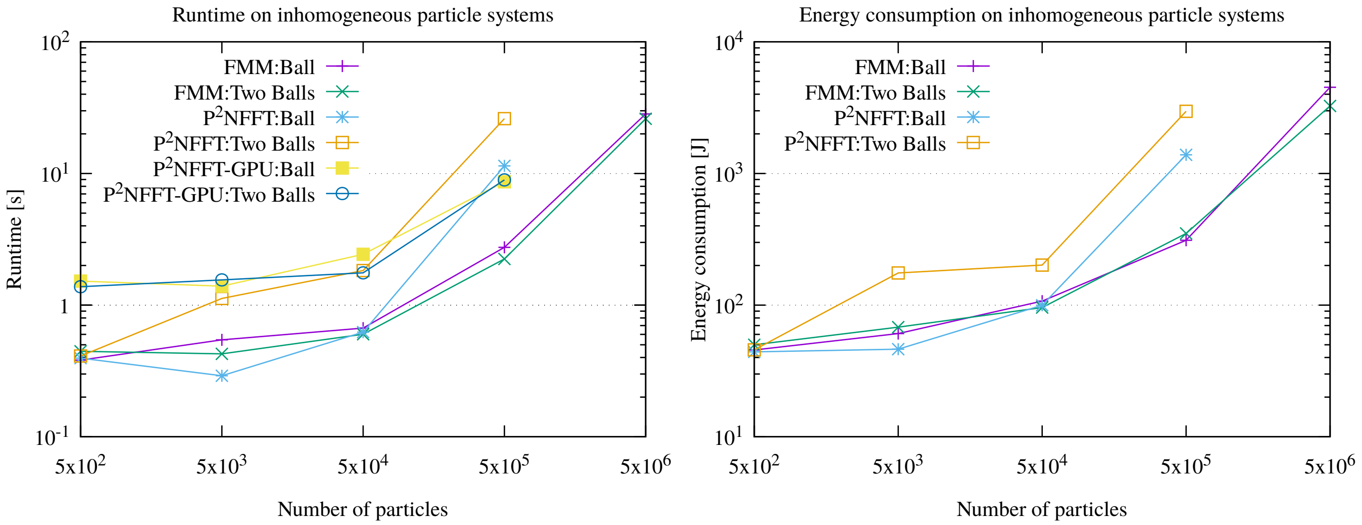


Figure 4. Parallel runtime (left) and energy measurements (right) for inhomogeneous systems with 56 MPI processes on the Haswell system and the Geforce GTX Titan Black.

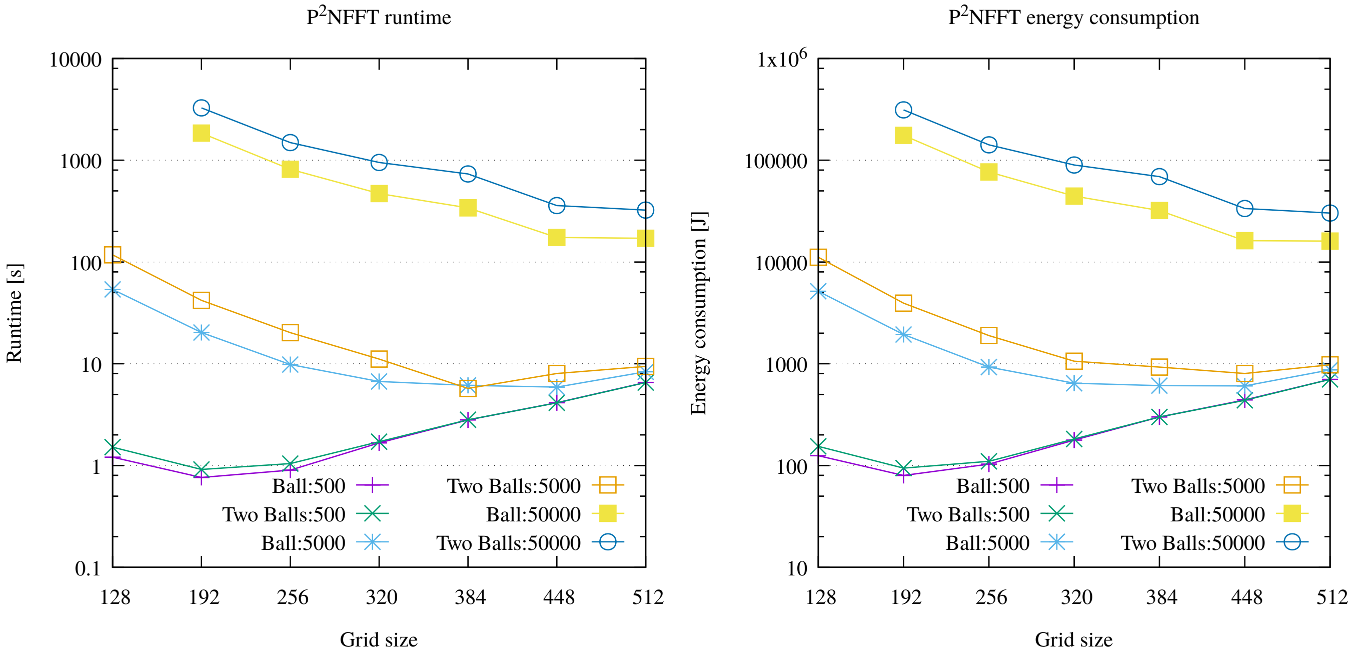


Figure 5. Parallel runtime (left) and energy measurements (right) for inhomogeneous systems with varying grid size with 56 MPI processes on the Haswell system.

solver depends on the distribution of the particle system, e.g. P^{2}NFFT for homogeneous distributions and FMM for inhomogeneous distributions (Two Balls particle system). For particle systems with more than 50,000 particles, the FMM solver is the best performing one if only CPUs are available. For a system with a less performing CPU, e.g., a single core CPU, the P^{2}NFFT OpenCL implementation on GPUs has the best performance for particle systems with more than 50,000 particles.

Usually, the characteristic of the particle input system is known before the execution starts, and measurement and evaluation results such as described above can help to start the most efficient simulation algorithm. However, there might be cases in which it is not *a priori* clear which characteristic the distribution of the input data might have. In these situations, it is possible to execute one time-step with each solver to measure the performance and then the results are compared to select the best one. In cases in which the solver specific parameters, e.g., the grid size, differ too much for the particle system distribution and size, the solvers have to be tested for several time-steps before the best one can be chosen.

In summary, the investigations of this article have shown that the performance results of the different simulation methods can differ for different particle distribution characteristics and different hardware, but that some behavior classes can be detected. This shows that there is a potential for designing tuning strategies based on a larger data basis.

VI. Related Work

Performance analysis and prediction of a particle simulation method was examined in [8]. As test system they used the SB-PRAM, a shared memory machine with up to 2048 processors.

Many implementations of the FMM approach invented in [1] exist and contain specific optimisations for the actual execution run. In [9] a parallel sorting for the particles in the particle systems is presented which improves the locality of interacting particles for computation on a distributed memory architecture. A more application specific optimization has been presented in [10], which introduces a method for automatic tuning of the FMM by selecting the optimal FMM tree depth based on an integrated performance prediction of the FMM computations.

The autotuning potential of particle simulation methods from the ScaFaCoS library are examined in [11] and [12]. In these articles, only one particle distribution is considered. In our article we consider different distributions of the particle systems and additionally examine an OpenCL solver. The OpenCL solver used is introduced and tested in [4].

In [13] and [14] autotuning strategies are introduced for different N-Body simulations on heterogeneous and hybrid CPU/GPU systems. The focus of these articles is on load balancing the GPUs, and thus less on CPU performance and energy consumption. The authors of [7] investigated two different ScaFaCoS solvers on different particle systems. We extended their work with a GPU solver and investigated different particle systems.

VII. Conclusions

The investigations of this article have shown that varying particle system distributions and sizes have a significant impact on the execution time and energy consumption. Using GPUs with the OpenCL implementation is useful when the CPU performance of the system is limited. The results show that some decisions can be made before runtime, but others, e.g., the solver specific parameters, have to be tested during runtime. For each particle simulation execution, the availability of hardware and the size of the particle system are fixed, but the distribution of the particles may change after some time steps, and thus a different particle simulation method could then be the best. Experiments have shown that the solver specific parameters, e.g., the grid size, have different optimal settings for different distributions and sizes. Thus, the performance results must be checked and compared to the values last checked. This can be done by monitoring during runtime. Our observations show that it is necessary to use both tuning approaches to tune runtime or energy consumption, an offline tuning to set the start parameters as well as possible, and an online approach to fine-tune the parameters, e.g., the solver specific parameters, and to respond to particle distribution changes.

Acknowledgment

This work is supported by the German Ministry of Science and Education (BMBF) project Selbstadaption für zeitschrittbasierte Simulationstechniken auf heterogenen HPC-Systemen (SeASiTe), Grant No. 01IH16012B.

References

[1] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations," *J. of Computational Physics*, vol. 73, pp. 325–348, 1987.

[2] M. Pippig and D. Potts, "Parallel three-dimensional nonequispaced fast fourier transforms and their application to particle simulation," *SIAM J. on Scientific Computing*, vol. 35, no. 4, pp. C411–C437, 2013. doi: 10.1137/120888478

[3] M. Bolten, F. Fahrenberger, R. Halver, F. Heber, M. Hofmann, I. Kabadshow, O. Lenz, M. Pippig, and G. Sutmann, "ScaFaCoS, C subroutine library," `http://scafacos.github.com/`. [Online]. Available: http://scafacos.github.com

[4] M. Hofmann, R. Kiesel, D. Leichsenring, and G. Rünger, "A hybrid cpu/gpu implementation of computationally intensive particle simulations using opencl," in *2018 17th International Symposium on Parallel and Distributed Computing (ISPDC)*, June 2018. doi: 10.1109/ISPDC2018.2018.00011 pp. 9–16.

[5] A. Arnold, F. Fahrenberger, C. Holm, O. Lenz, M. Bolten, H. Dachsel, R. Halver, I. Kabadshow, F. Gähler, F. Heber, J. Iseringhausen, M. Hofmann, M. Pippig, D. Potts, and G. Sutmann, "Comparison of scalable fast methods for long-range interactions," *Physical Review E*, vol. 88, p. 063308, 2013.

[6] J. M. Hammersley, "Monte carlo methods for solving multivariable problems," *Annals of the New York Academy of Sciences*, vol. 86, no. 3, pp. 844–874, 1960. doi: 10.1111/j.1749-6632.1960.tb42846.x. [Online]. Available: https://nyaspubs.onlinelibrary.wiley.com/doi/abs/10.1111/j.1749-6632.1960.tb42846.x

[7] M. Pippig and D. Potts, "Particle simulation based on nonequispaced fast fourier transforms," 01 2011, pp. 131 – 158.

[8] T. Rauber, G. Rünger, and C. Scholtes, "Execution Behavior Analysis and Performance Prediction for a Shared-Memory Implementation of an Irregular Particle Simulation Method," *Simulation: Practice and Theory*, vol. 6, no. 7, pp. 665–687, 1998. doi: 10.1016/S0928-4869(98)00006-8

[9] H. Dachsel, M. Hofmann, and G. Rünger, "Library Support for Parallel Sorting in Scientific Computations," in *Proceedings of the 13th International Euro-Par Conference*, ser. LNCS, vol. 4641. Springer, August 2007. doi: 10.1007/978-3-540-74466-5_73. ISBN 978-3-540-74465-8 pp. 695–704.

[10] H. Dachsel, M. Hofmann, J. Lang, and G. Rünger, "Automatic Tuning of the Fast Multipole Method Based on Integrated Performance Prediction," in *Proceedings of the 14th IEEE International Conference on High Performance Computing and Communications (HPCC-2012)*. IEEE, Juni 2012. doi: 10.1109/HPCC.2012.88. ISBN 978-1-4673-2164-8 pp. 617–624.

[11] N. Kalinnik, R. Kiesel, T. Rauber, M. Richter, and G. Rünger, "On the Autotuning Potential of Time-stepping methods from Scientific Computing," in *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems (FedCSIS 2018), 11th Workshop on Computer Aspects of Numerical Algorithms (CANA'18)*, vol. 15. ACSIS, September 2018. doi: 10.15439/2018F169. ISSN 2300-596 pp. 329–338.

[12] M. Hofmann, R. Kiesel, and G. Rünger, "Energy and Performance Analysis of Parallel Particle Solvers from the ScaFaCoS Library," in *Proceedings of the 2018 ACM/SPEC International Conference on Performance Engineering (ICPE 2018)*. ACM, April 2018. doi: 10.1145/3184407.3184409. ISBN 978-1-4503-5095-2 pp. 88–95.

[13] R. Yokota and L. Barba, "Hierarchical N-body Simulations with Autotuning for Heterogeneous Systems," *Computing in Science Engineering*, vol. 14, no. 3, pp. 30–39, May 2012. doi: 10.1109/MCSE.2012.1

[14] M. Holm, S. Engblom, A. Goude, and S. Holmgren, "Dynamic Autotuning of Adaptive Fast Multipole Methods on Hybrid Multicore CPU and GPU Systems," *SIAM Journal on Scientific Computing*, vol. 36, no. 4, pp. C376–C399, Jan. 2014. doi: 10.1137/130943595. [Online]. Available: https://epubs.siam.org/doi/abs/10.1137/130943595

6th International Conference on Cryptography and Security Systems

CRYPTOGRAPHY and security systems are two fields of security research that strongly interact and complement each other. The International Conference on Cryptography and Security Systems (CSS) is a forum of presentation of theoretical, applied research papers, case studies, implementation experiences as well as work-in-progress results in these two disciplines.

TOPICS

The main topics of interests include:

- network security
- cryptography and data protection
- peer-to-peer security
- security of wireless sensor networks
- security of cyber physical systems
- security of Internet of Things solutions
- heterogeneous networks security
- privacy-enhancing methods
- covert channels
- steganography and watermarking for security applications
- cryptographic protocols
- security as quality of service, quality of protection
- data and application security, software security
- security models, evaluation, and verification
- formal methods in security
- trust and reputation models
- reputation systems for security applications
- intrusion tolerance
- system surveillance and enhanced security
- cybercrime: threats and countermeasures
- 5G Security
- DDoS attacks: detection and mitigation
- Security of Smart Grid systems

EVENT CHAIRS

- **Kotulski, Zbigniew,** Warsaw University of Technology, Faculty of Electronics and Information Technology, Institute of Telecommunications, Department of Cybersecurity, Poland
- **Ksiezopolski, Bogdan,** Maria Curie-Sklodowska Univerity, Faculty of Mathematics, Physics and Computer Science, Institute of Computer Science, Department of Cybersecurity and Polish-Japanese Academy of Information Technology, Poland

PROGRAM COMMITTEE

- **Cabaj, Krzysztof,** Institute of Computer Science, Warsaw University of Technology, Poland
- **Caviglione, Luca,** National Research Council (CNR), Italy
- **Cheng, Shin-Ming,** National Taiwan University of Science and Technology, Taiwan
- **Courtois, Nicolas T.,** University College London, United Kingdom
- **Domingos, Maria Dulce Pedroso,** Universidade de Lisboa, Portugal
- **El Fray, Imed,** Warsaw University of Life Sciences, Faculty of Applied Informatics and Mathematics, Poland
- **Gajewski, Piotr,** Military University of Technology, Poland
- **Górski, Janusz,** Gdańsk University of Technology, Poland
- **Grocholewska-Czuryło, Anna,** Poznan University of Technology, Poland
- **Gutierrez, Jaime,** Universidad de Cantabria, Spain
- **Hyla, Tomasz,** West Pomeranian University of Technology, Poland
- **Kotenko, Igor,** St.Petersburg Institute for Informatics and Automation, Russia
- **Kula, Mieczysław,** University of Silesia, Poland
- **Mauw, Sjouke,** University of Luxembourg, Luxembourg
- **Mazurczyk, Wojciech,** Warsaw University of Technology, Poland
- **Memmi, Gérard,** Telecom ParisTech, France
- **Nielek, Radosław,** Polish-Japanese Academy of Information Technology
- **Pejaś, Jerzy,** West Pomeranian University of Technology, Poland
- **Pieprzyk, Josef,** Queensland University of Technology, Australia
- **Piotrowski, Zbigniew,** Military University of Technology, Poland
- **Respício, Anna,** Universidade de Lisboa, Portugal
- **Ryan, Peter Y A,** University of Luxembourg, Luxembourg
- **Seredyński, Franciszek,** Cardinal Wyszyński University in Warsaw, Poland
- **Szałachowski, Paweł,** SUTD, Singapore
- **Tiplea, Ferucio,** Alexandru Ioan Cuza University of Iasi, Romania
- **Ustimenko, Vasyl,** Marie Curie-Sklodowska University, Poland
- **Wydra, Michał,** Lublin University of Technology, Poland

Position Papers of the Federated Conference on Computer Science and Information Systems pp. 41–46
DOI: 10.15439/2019F343
ISSN 2300-5963 ACSIS, Vol. 19

Expanding graphs of the Extremal Graph Theory and expanded platforms of Post Quantum Cryptography

Vasyl Ustymenko
Maria Curie-Skłodowska University
pl. Marii Curie-Skłodowskiej 1
20-031 Lublin, Poland
Email: vasylustimenko@yahoo.pl

Urszula Romańczuk-Polubiec
Independent Researcher, Poland
Email: urszula_romanczuk@yahoo.pl

Aneta Wróblewska
Maria Curie-Skłodowska University
pl. Marii Curie-Skłodowskiej 1
20-031 Lublin, Poland
Email: awroblewska@hektor.umcs.lublin.pl

***Abstract*—Explicit constructions in Extremal Graph Theory give appropriate lower bounds for Turan type problems. In the case of prohibited cycles, the explicit constructions can be used for various problems of Information Security. We observe recent applications of algebraic constructions of regular graphs of large girth and graphs with large cycle indicator to Coding Theory and Cryptography. In particular, we present a new multivariate platforms of postquantum Non-commutative Cryptography defined in graph theoretical terms.**

***Index Terms*—graphs of large girth, graphs of large cycle indicator, graph based stream ciphers, multivariate cryptography, non-commutative cryptography**

I. Some definitions of Extremal Graph Theory

THE missing definitions of graph-theoretical concepts in the case of simple graphs which appears in this paper can be found in [1]. All graphs we consider are simple ones, i. e. undirected without loops and multiple edges. When it is convenient, we shall identify Γ with the corresponding antireflexive binary relation on $V(\Gamma)$, i.e. $E(\Gamma)$ is a subset of $V(\Gamma) \times V(\Gamma)$. The *girth* of a graph Γ, denoted by $g = g(\Gamma)$, is the length of the shortest cycle in Γ. The *diameter* $d = d(\Gamma)$ of the graph Γ is the maximal length of the shortest pass between its two vertices.

Let $g_x = g_x(\Gamma)$ be the length of the minimal cycle through the vertex x from the set $V(\Gamma)$ of vertices in graph Γ. We refer to $Cind(\Gamma) = max\,\{g_x,\ x \in V(\Gamma)\}$ as *cycle indicator* of the graph Γ. The family Γ_i of connected k-regular graphs of constant degree is a *family of small world graphs*, if $d(\Gamma_i) \leq c\, \log_k(v_i)$, for some constant c, $c > 0$. Recall that family of regular graphs Γ_i of degree k and increasing order v_i is a *family of graphs of large girth*, if $g(\Gamma_i) \geq c \log_k(v_i)$, for some independent constant c, $c > 0$. We refer to the family of regular simple graphs Γ_i of degree k and order v_i as a *family of graphs of large cycle indicator*, if $Cind(\Gamma_i) \geq c \log_k(v_i)$ for some independent constant c, $c > 0$.

Notice that for vertex -transitive graph its girth and cycle indicator coincide. Defined above families plays an important role in Extremal Graph Theory, Theory of LDPC codes and Cryptography (see [2] and further references).

II. The algebraic graphs $A(n, \mathbb{K})$ and $D(n, \mathbb{K})$, some results and open questions

Below we consider the family of graphs $A(n, \mathbb{K})$ and $D(n, \mathbb{K})$, respectively where $n > 5$ is a positive integer and $\mathbb{K}$ is a commutative ring. In the case of $\mathbb{K} = \mathbb{F}_q$, we denote $A(n, q)$ and $D(n, q)$, respectively. We define these graphs as homomorphic images of infinite bipartite graphs $A(\mathbb{K})$ and $D(\mathbb{K})$ for which partition sets P and L formed by two copies of Cartesian power $\mathbb{K}^{\mathbb{N}}$, where $\mathbb{K}$ is the commutative ring and $\mathbb{N}$ is the set of positive integer numbers. Elements of P will be called points and those of L lines. To distinguish points from lines we use parentheses and brackets. If $x \in V$, then $(x) \in P$ and $[x]$ ∈L. The description is based on the connections of these graphs with Kac-Moody Lie algebra with extended diagram A_1.

The vertices of $D(\mathbb{K})$ are infinite dimensional tuples over $\mathbb{K}$. We write them in the following way $(p) = (p_{0,1}, p_{1,1}, p_{1,2}, p_{2,1}, p_{2,2}, p'_{2,2}, p_{2,3}, \ldots, p_{i,i}, p'_{i,i}, p_{i,i+1}, p_{i+1,i}, \ldots)$, $[l] = [l_{1,0}, l_{1,1}, l_{1,2}, l_{2,1}, l_{2,2}, l'_{2,2}, l_{2,3}, \ldots, l_{i,i}, l'_{i,i}, l_{i,i+1}, l_{i+1,i}, \ldots]$. We assume that almost all components of points and lines are zeros. The condition of incidence of point (p) and line $[l]$, i.e. $(p)\ I\ [l]$, can be written via the list of equations below.

$$\left\{\begin{array}{l} l_{i,i} - p_{i,i} = l_{1,0}p_{i-1,i}, \\ l'_{i,i} - p'_{i,i} = p_{0,1}l_{i,i-1}, \\ l_{i,i+1} - p_{i,i+1} = p_{0,1}l_{i,i}, \\ l_{i+1,i} - p_{i+1,i} = l_{1,0}p'_{i,i}. \end{array}\right. \tag{1}$$

This four relations are defined for $i \geq 1$, with $p'_{1,1} = p_{1,1}$, $l_{1,1} = l_{1,1}$.

Similarly we define graphs $A(\mathbb{K})$ on the vertex set consisting of points and lines $(p) = (p_{0,1}, p_{1,1}, p_{1,2}, p_{1,2}, p_{2,2}, p_{2,3}, \ldots, p_{i,i}, p_{i,i+1}, \ldots)$, $[l] = [l_{1,0}, l_{1,1}, l_{1,2}, l_{2,1}, l_{2,2}, l_{2,3}, \ldots, l_{i,i}, l_{i,i+1}, \ldots]$ such that point (p) is incident with the line $[l]$, i.e. $(p)\ I\ [l]$, if the following relations between their coordinates hold:

$$\left\{\begin{array}{l} l_{i,i} - p_{i,i} = l_{1,0}p_{i-1,i}, \\ l_{i,i+1} - p_{i,i+1} = p_{0,1}l_{i,i}. \end{array}\right. \tag{2}$$

It is clear that the set of indices $A = \{(1,0);\ (0,1);\ (1,1);\ (1,2);\ (2,2);\ (2,3);\ \ldots,\ (i-1,i);\ (i,i);\ \ldots\}$ is a subset in $D = \{(1,0);\ (0,1);\ (1,1);\ (1,2);\ (2,2);\ (2,2)';\ \ldots;\ (i-1,i);\ (i,i-1);\ (i,i);\ (i,i)';\ \ldots\}$. Points and lines of $D(\mathbb{K})$ are functions from $\mathbb{K}^{D-\{(1,0)\}}$ and $\mathbb{K}^{D-\{(0,1)\}}$ and their restrictions on $A - \{(1,0)\}$ and $A - \{(0,1)\}$ define homomorphism Ψ of graph $D(\mathbb{K})$ onto $A(\mathbb{K})$. For each positive integer $m \geq 2$ we consider subsets $A(m)$ and $D(m)$ containing first $m+1$ elements of A and D with respect to the above orders. Restrictions of points and lines of $D(\mathbb{K})$ onto $D(m) - \{(1,0)\}$ and $D(m) - \{(0,1)\}$ define graph homomorphism ${}^{D}\Delta(m)$ with image denoted as $D(n,\mathbb{K})$. Similarly restrictions of points and lines of $A(\mathbb{K})$ onto $A(m) - \{(1,0)\}$ and $A(m) - \{(0,1)\}$ defines homomorphism ${}^{A}\Delta(m)$ of graph $A(\mathbb{K})$ onto graph denoted as $A(m,\mathbb{K})$.

We also consider the map $\Delta(m)$ on vertices of graph $D(m,\mathbb{K})$ sending its point $(p) \in \mathbb{K}^{|D(m)-\{(1,0)\}|}$ to its restriction into $D(m) \cap A - \{(1,0)\}$ and its line $[l] \in K^{|D(m)-\{(0,1)\}|}$ to its restriction onto $D(m) \cap A - \{(0,1)\}$. This map is homomorphism of $D(m,\mathbb{K})$ onto $A(n,\mathbb{K})$, $n = |D(m) \cap A| - 1$. Graph $D(q) = D(\mathbb{F}_q)$ is q-regular forest. Its quotients $D(n,q)$ are edge transitive graphs. So their connected components are isomorphic. Symbol $CD(n,q)$ stands for the graph which is isomorphic to one of such connected components. Family $CD(n,q)$, $n = 2,3,\ldots$ is a family of large girth for each parameter q, $q > 2$ (see [3] and further references). The question "*Whether or not $CD(n,q)$ is a family of small world graphs?*" is still open. Graph $A(q)$, $q > 2$ is a q-regular tree. Graphs $A(n,q)$ are not vertex transitive. They form a family of graphs with large cycle indicator, which is q-regular family of small world graphs [4]. The question "*Whether or not $A(n,q)$, $n = 2,3,\ldots$ is a family of large girth?*" is still open. Graphs $CD(n,q)$ and $A(n,q)$ are expanding graphs (see [10], [20], [45], [46]) with spectral gap $q - 2\sqrt{q}$.

Groups $GD(n,\mathbb{K})$ and $GA(n,\mathbb{K})$ of cubical transformations of affine space $\mathbb{K}^n$ associated with graphs $D(n,\mathbb{K})$ and $A(n,\mathbb{K})$ are interesting objects of algebraic transformation group theory because of composition of two maps of degree 3 for vast majority of pairs will have degree 9. Applications of these groups to Symmetric Cryptography are observed in [5], [6], they are used in Multivariate Cryptography (see [7]-[13]). Papers [14],[15], [16] devoted to applications of these groups as so called platforms of Non-commutative Cryptography (see [17]). Cryptographic applications of other graphs are observed in [18].

III. On linguistic and extremal graphs and stable nonlinear subgroups of affine Cremona group

All graphs defined in section 2 belong to class L of linguistic graphs $\Gamma = \Gamma(\mathbb{K})$ of type $(1,1,n-1)$, $n \in \mathbb{N}$ or $n = \infty$ defined over commutative ring $\mathbb{K}$ which contains bipartite graphs with the point set $P_n = \mathbb{K}^n$ and line set $L_n = \mathbb{K}^n$ such that $(p) = (p_1,p_2,\ldots,p_n) \in P_n$ and $[l] = [l_1,l_2,\ldots,l_n] \in L_n$ form an edge of Γ if the following conditions holds

$$\begin{cases} {}^{2}\,bl_2 - {}^{2}\,ap_2 = {}^{2}\,f(p_1,l_1), \\ {}^{3}\,bl_3 - {}^{3}\,ap_3 = {}^{3}\,f(p_1,p_2,l_1,l_2), \\ \vdots \\ {}^{n}\,bl_n - {}^{n}\,ap_n = {}^{n}\,f(p_1,p_2,\ldots,p_{n-1},l_1,l_2,\ldots,l_{n-1}), \end{cases} \tag{3}$$

where ${}^{i}\,a$ and ${}^{i}\,b$, $i \geq 2$ are elements of multiplicative group $\mathbb{K}^*$ (see [43] or [44]) and ${}^{i}\,f$ are multivariate polynomials. We define colours $\rho((p))$ and $\rho([l])$ of the point (p) and the line $[l]$ as their first coordinates p_1 and l_1. We introduce well defined the *neighbour operator* $N(v,a)$ of computing the neighbour of vertex v of colour $a \in \mathbb{K}$ and the *colour jump operator* $J(v,a)$ sending point or line $v = (v_1,v_2,\ldots,v_n)$ to $u = (a,v_2,v_3,\ldots,v_n)$.

Let $S(\mathbb{K}^n)$ stands for the Cremona semigroup of polynomial transformations of free module $\mathbb{K}^n$ and $C(\mathbb{K}^n)$ be affine Cremona group of invertible elements of $S(\mathbb{K}^n)$ with the polynomial inverse. These algebraic structures are important objects of algebraic geometry. One of the difficult problem is about constructions of families of stable subgroups G_n of $C(\mathbb{K}^n)$ (or semigroup S_n of $S(\mathbb{K}^n)$), i.e. groups of polynomial transformation with maximal degree equals to constant c. Notice that for the majority of pair $f,g \in C(\mathbb{K}^n)$ of degrees r and s their composition has degree rs. So this problem is difficult, it has strong cryptographical motivations.

We consider totality $St(\mathbb{K})$ of strings of kind $(f_1,f_2,\ldots,f_k)$, where $f_i \in \mathbb{K}[x]$. We will identify polynomial f and the map $x \to f(x)$ from $S(\mathbb{K})$. The product of two chains $(f_1,f_2,\ldots,f_k)$ and $(g_1,g_2,\ldots,g_t)$ is the chain $(f_1,f_2,\ldots,f_k,g_1(f_k),g_2(f_k),\ldots,g_t(f_k))$. Empty string is the unity of semigroup $St(\mathbb{K})$. In fact $St(\mathbb{K})$ is a semidirect product of a free semigroup over the alphabet $\mathbb{K}[x]$ and Cremona semigroup $S(\mathbb{K})$. We refer to $St(\mathbb{K})$ as semigroup of polynomial strings. Let $St'(\mathbb{K})$ stands for the semigroup of strings of even length from $St(\mathbb{K})$ and $\sum(\mathbb{K})$ be subsemigroups of strings of even length with coordinates of kind $x+c$, $c \in \mathbb{K}$.

In the case of linguistic graph $\Gamma = \Gamma(\mathbb{K})$ of type $(1,1,n-1)$ the path consisting of its vertices $v_0,v_1,v_2,\ldots,v_k$ is uniquely defined by initial vertex v_0, and colours $\rho(v_i)$, $i = 1,2,\ldots,k$ of other vertices from the path. We can consider graph $\Gamma' = \Gamma(\mathbb{K}[x_1,x_2,\ldots,x_n])$ defined by the same with Γ equations but over the commutative ring $\mathbb{K}[x_1,x_2,\ldots,x_n]$. So the following symbolic computation can be defined. Take the symbolic point $x = (x_1,x_2,\ldots x_n)$, where x_i are generic variables of $\mathbb{K}[x_1,x_2,\ldots,x_n]$ and polynomial string $C \in St'(\mathbb{K})$ which is a tuple of polynomials $f_1,f_2,\ldots,f_k$ from $\mathbb{K}[x_1]$ with even parameter k $(x = x_1)$. Form the path of vertices $v_0 = x$ and $\rho(v_0) = x_1$, v_1 such that $v_1\,I\,v_0$ and $\rho(v_1) = f_1(x_1)$, v_2 such that $v_2\,I\,v_1$ and $\rho(v_2) = f_2(x_1)$, …, v_k such that $v_k\,I\,v_{k-1}$ and $\rho(v_k) = f_k(x_1)$. We choose parameter k as even number. So v_k is the point from the partition set $\mathbb{K}[x_1,x_2,\ldots,x_n]$ of the graph Γ'.

We notice that the computation of each coordinate of v_i depending on variables x_1, x_2, …, x_n and polynomials

f_1, f_2, $\ldots$, f_k needs only arithmetical operations of addition and multiplication. As it follows from the definition of linguistic graph final vertex v_k (point) has coordinates $(h_1(x_1),\ h_2(x_1,x_2),\ h_3(x_1,x_2,x_3),\ldots,\ h_n(x_1,x_2,\ldots,x_n))$, where $h_1(x_1) = f_k(x_1)$. Let us consider the map ${}^{\Gamma}H(C)$: $x_i \to h_i(x_1,x_2,\ldots,x_n)$, $i = 1,2,\ldots,n$, which corresponds to polynomial string C.

Proposition 1: The map ${}^{\Gamma}\eta : C \to {}^{\Gamma}H(C)$ is a homomorphism of $St'(\mathbb{K})$ into Cremona semigroup $S(\mathbb{K}^n)$.

More general form of this statement is proven in [20]. We refer to ${}^{\Gamma}\eta$ as the *linguistic compression map*. If $\mathbb{K}$ is finite then the map converts totality of potentially infinite strings into finite semigroup.

Theorem 2: If Γ is one of graphs $D(n,\mathbb{K})$ and $A(n,\mathbb{K})$, then ${}^{\Gamma}\eta(\sum(\mathbb{K}))$ is stable subgroup of $C(\mathbb{K}^n)$ of degree 3.

We denote ${}^{\Gamma}\eta(\sum(\mathbb{K}))$ for $\Gamma = D(n,\mathbb{K})$ and $\Gamma = A(n,\mathbb{K})$ as $GD(n,\mathbb{K})$ and $GA(n,\mathbb{K})$. These groups were already used in all cryptographical applications of graphs $D(n,\mathbb{K})$ and $A(n,\mathbb{K})$.

Proposition 3: Homomorphisms δ of $D(n,\mathbb{K})$ onto $A(m,\mathbb{K})$, $n > m$ described in section 2 induces homomorphism of $GD(n,\mathbb{K})$ onto $GA(m,\mathbb{K})$, $n > m$.

IV. On linguistic graphs and expansions of stable nonlinear subgroups of affine Cremona group

Let $St'(\mathbb{K})$ stands for the semigroup of strings of even length from $St(\mathbb{K})$ and $\sum(\mathbb{K})$ be subsemigroups of strings of even length with coordinates of kind $x + c$, $c \in \mathbb{K}$.

In the case of linguistic graph $\Gamma = \Gamma(\mathbb{K})$ of type $(1,1,n-1)$ the sequence of even length $k = 2r$ consisting of initial vertex v_0 and $v_1 = J(v_0,a_1)$, $v_2 = N(v_1,b_1)$, $v_3 = J(v_2,a_2)$, $v_4 = N(v_3,b_2)$, $\ldots$, $v_{k-1} = J(v_{k-2},a_r)$, $v_k = N(v_{k-1},b_r)$ is uniquely defined by initial vertex v_0, and colours parameter $(a_1,a_2,\ldots,a_r)$ and $(b_1,b_2,\ldots,b_r)$. We can consider graph $\Gamma' = \Gamma(\mathbb{K}[x_1,x_2,\ldots,x_n])$ defined by the same with Γ equations but over the commutative ring $\mathbb{K}[x_1,x_2,\ldots,x_n]$. So the following symbolic computation can be defined. Take the symbolic point $x = (x_1,x_2,\ldots,x_n)$, where x_i are generic variables of $\mathbb{K}[x_1,x_2,\ldots,x_n]$ and polynomial string $C \in St'(\mathbb{K})$, which is a tuple of polynomials f_1, f_2, $\ldots$, f_k from $\mathbb{K}[x_1]$ with even parameter k ($x = x_1$). Form the path of vertices $v_0 = x$, v_1 such that $v_1 = J(v_0,f_1(x_1))$, $v_2 = N(v_1,f_2(x_1))$, $v_3 = J(v_2,f_3(x_1))$, $v_4 = N(v_3,f_4(x_1))$, $\ldots$, $v_{k-1} = J(v_{k-2},f_{k-1}(x_1))$, $v_k = N(v_{k-1},f_k(x_1))$ and $\rho(v_2) = f_2(x_1)$. We choose parameter k as even number. So v_k is the point from the partition set $\mathbb{K}[x_1,x_2,\ldots,x_n]$ of the graph Γ'. We notice that the computation of each coordinate of v_i depending on variables x_1, x_2, $\ldots$, x_n and polynomials $f_1, f_2, \ldots, f_k$ needs only arithmetical operations of addition and multiplication. As it follows from the definition of linguistic graph final vertex v_k (point) has coordinates $(h_1(x_1),\ h_2(x_1,x_2),\ h_3(x_1,x_2,x_3),\ \ldots,\ h_n(x_1,x_2,\ldots,x_n))$, where $h_1(x_1) = f_k(x_1)$. Let us consider the map ${}^{\Gamma}L(C)$: $x_i \to h_i(x_1,x_2,\ldots,x_n)$, $i = 1,2,\ldots,n$, which corresponds to polynomial string C.

Proposition 4: The map ${}^{\Gamma}\mu : C \to {}^{\Gamma}L(C)$ is a homomorphism of $St'(\mathbb{K})$ into Cremona semigroup $S(\mathbb{K}^n)$.

More general form of this statement is proven in [Us pust].

Theorem 5: If Γ is one of graphs $D(n,\mathbb{K})$ and $A(n,\mathbb{K})$ then ${}^{\Gamma}\mu(\sum(\mathbb{K}))$ is stable subgroup of $C(\mathbb{K}^n)$ of degree 3.

We denote ${}^{\Gamma}\mu(\sum(\mathbb{K}))$ for $\Gamma = D(n,\mathbb{K})$ and $\Gamma = A(n,\mathbb{K})$ as $JD(n,\mathbb{K})$ and $JA(n,\mathbb{K})$. As it follows from definitions $JD(n,\mathbb{K}) > GD(n,\mathbb{K})$ and $JA(n,\mathbb{K}) > GA(n,\mathbb{K})$.

Proposition 6: Homomorphisms δ of $D(n,\mathbb{K})$ onto $A(m,\mathbb{K})$, $n > m$ described in section 2 induces homomorphism of $JD(n,\mathbb{K})$ onto $JA(m,\mathbb{K})$, $n > m$.

V. On cryptosystems based on new multivariate platforms of Non-commutative Cryptography

Non-commutative cryptography appeared with attempts to apply Combinatorial group theory to Information Security. If G is noncom-mutative group then correspondents can use conjugations of elements involved in protocol, some algorithms of this kind were suggested in [22], [23], [24], [25], where group G is given with the usage of generators and relations. Security of such algorithms is connected to Conjugacy Search Problem (CSP) and Power Conjugacy Search Problem (PCSP), which combine CSP and Discrete Logarithm Problem and their generalizations. Currently Non-commutative cryptography is essentially wider than group based cryptography. It is an active area of cryptology, where the cryptographic primitives and systems are based on algebraic structures like groups, semigroups and noncommutative rings (see [26]-[33]). This direction of security research has very rapid development (see [34], [35] and further references in these publications).

One of the earliest applications of a non-commutative algebraic structures for cryptographic purposes was the usage of braid groups to develop cryptographic protocols. Later several other non-commutative structures like Tompson groups and Grigorchuk groups have been identified as potential candidates for cryptographic post quantum applications. The standard way of presentations of groups and semigroups is the usage of generators and relations (Combinatorial Group Theory). Semigroup based cryptography consists of general cryptographic schemes defined in terms of wide classes of semigroups and their implementations for chosen semigroup families (so called platform semigroups).

The paper is devoted to some research on the intersection of Non Commutative and Multivariate Cryptographies. We try to use some abstract schemes in terms of Combinatorial Semigroup Theory for the implementation with platforms which are semigroups and groups of polynomial transformations of free modules $\mathbb{K}^n$ where $\mathbb{K}$ is commutative ring.

The most popular form of Multivariate cryptosystem is the usage of a single very special map f in a public key mode. First examples were based on families of quadratic bijective transformation f_n (see [36], [37], [38]), such choice implies rather fast encryption process.

Some of recent applications of extremal graphs are connected with other aspects of Multivariate cryptography when some subsemigroup of affine Cremona semigroup of all polynomial transformations is used instead of a single transformation. Notice that the implementation of the idea to use several multivariate generators in its standard form has to overcome essential difficulties. At first glance this idea looks as unrealistic one because of com-position of two maps of degree r and s taken in "general position " will be a transformation of degree rs. So in majority of cases $deg(F) = d$, $d > 1$ implies very fast growth of function $d(r) = deg(F^r)$. Of course in the case of generator in common position not only degree but a density (total number of monomial terms of the map in its standard forms) grows exponentially.

So we have to search for special conditions on subsemigroup of affine Cremona group which guarantee the polynomial complexity of procedure to compute the composition of several elements from subsemigroup. Such conditions can define a basis of Noncommutative Multivariate Cryptography. The stability condition on subsemigroup which we discussed above is one of them. Recently we noticed that condition of minimal possible density (each f_i in standard form has density 1) also guarantee efficiency of computations (see [19]). The idea to combine representative of stable group (for example $GD(n, \mathbb{K})$ or $GA(n, \mathbb{K})$) and non-bijective transformation of minimal density is used in [40] and [41] for the construction of new postquantum cryptosystems.

The abstract schemes of Nonlinear Cryptography has to be modified to work with stable subsemigrouos or subsemigroups of minimal density. The following TAHOMA CRYPTOSYSTEM on stable transformations were suggested in [15].

Let $\mathbb{K}$ be a commutative ring, stable subgroups ${}^nG$ of $S(\mathbb{K}^n)$ act naturally on Kn and ${}^mS(n, \mathbb{K})$ be a subgroup of $S(\mathbb{K}^m)$ such that there is a tame homo-morphisn $\Delta = \Delta(m, n)$ of ${}^mS(n, \mathbb{K})$ onto ${}^nG$. We assume that $m = m(n)$ where $m > n$. Alice takes b_1, b_2, $\ldots$, b_s, $s > 1$ from ${}^mS(n, \mathbb{K})$ and a_1, a_2, $\ldots$, a_s, where $a_i = \Delta(b_i)^{-1}$. She takes $g \in C(\mathbb{Q}^m)$ and $h \in C(\mathbb{T}^n)$ where $\mathbb{Q}$ and $\mathbb{T}$ are extensions of the commutative ring $\mathbb{K}$ and forms pairs $(g_i, h_i) = (g^{-1}b_ig, h^{-1}a_ih)$, $i = 1, 2, \ldots, s$ and sends them to Bob. We assume that $g = g'T$, $h = h'T'$ where semigroup $< g',^m S(n, \mathbb{K}) >$ generated by g' and elements of ${}^mS(n, \mathbb{K})$ and group $< h', G >$ are stable semigroups of degree d and $T \in AGL_n(\mathbb{T}), T' \in AGL_m(\mathbb{Q})$.

As in the previous algorithm Bob writes the word $w(z_1, z_2, \ldots, z_s)$ in the alphabet z_1, z_2, $\ldots$, z_s together with the reverse word $w'(z_1, z_2, \ldots, z_s)$ formed by characters of w written in the reverse order. He computes element $b = w(g_1, g_2, \ldots, g_s)$ via specialization $z_i = g_i$ and $a = w'(h_1, h_2, \ldots, h_s)$ via specialization $z_i = h_i$. Bob keeps a for himself and sends b to Alice.

She computes a^{-1} as $h^{-1}\Delta(gbg^{-1})h$. Alice writes her message $(p_1, p_2, \ldots, p_n)$ from $\mathbb{T}^n$ and sends ciphertext $a^{-1}(p_1, p_2, \ldots, p_n)$ to Bob. He decrypts with his function a. Symmetrically Bob sends his ciphertext $a(p_1, p_2, \ldots, p_n)$ to Alice and she decrypts with a^{-1} (see [21]). Let ${}^nTC(\mathbb{K}, \mathbb{T}, \mathbb{Q})$ stand for Tahoma cryptosystem as above.

Paper [16] is devoted to implementations of Affine Tahoma scheme with platforms of cubical stable groups $GD(n, q)$ and $GA(n, q)$. They were defined via families of linguistic graphs which form projective limits and the standard homomorphisms between two members of this sequences. So we have pairs (G_n, Δ_n), where $G_n < S(\mathbb{K}^n)$, Δ_n is a homomorphism of G_n onto G_m, $m = m(n)$ such that projective limits $\lim(G_n)$, $n \to \infty$, and $\lim(\Delta(G_n))$, $n \to \infty$, coincide with the same infinite transformation group G.

The article [42] is devoted to another computer experiment with the new platform which uses the same groups G_n but different tame homomorphisms η_n. In the new scheme $\lim(G_n)$, $n \to \infty$, equals to G, but $\lim(\eta_n(G_n))$, $n \to \infty$, coincides with the image of homomorphism of G with an infinite kernel.

We believe that option to vary tame homomorphisms in the chosen sequence of semigroup makes the task of cryptanalytic much more difficult.

Extensions of groups $GD(n, \mathbb{K})$ and $GA(n, \mathbb{K})$ to new essentially large groups $JD(n, \mathbb{K})$ and $JA(n, \mathbb{K})$ allows to use new groups and defined above homomorphism between them for new more secure realisations of Tahoma schemes. Obviously WP problem is harder un the case of generators freely chosen from the larger group.

Other advantage of the implememtation of Tahoma cryptosystems with groups ${}^m\, S(m, \mathbb{K}) = JD(m, \mathbb{K})$ and ${}^n\, G = JA(n, \mathbb{K})$ and homomorphism δ of Proposition 6 between them is much faster computation of generator b_i as images of words w_i under ${}^\Gamma\, \mu$, $\Gamma = D(m, \mathbb{K})$ and $a_i = \delta(b_i)^{-1}$ in comparison with case of $GD(m, \mathbb{K})$ and $GA(n, \mathbb{K})$. To make comparison fair we have to assume that length of words from $St'(\mathbb{K})$ is fixed. Currenly we are working on detailed complexity estimates and investigation of statistical mixing properties on the base of computer simulation.

VI. Conclusion

We present a short survey of our recent algorithms on applications of Extremal Expander Graphs to Cryptography which appear after publication of [2] at memorial Erdos conference and announce the theorem about new explicitly constructed families of stable groups. The main added instruments are

(1) usage of non-bijective transformations defined in terms of algebraic graphs for the constructions of new stream ciphers and public key cryptosystems,
(2) usage of compositions of stable transformation of affine space $\mathbb{K}^n$ and transformation of minimal possible density (n),
(3) work on the bridge between Multivariate Cryptography and Non-commutative Cryptography, modification of schemes of protocols and El Gamal cryptosystems for platforms of elements of affine Cremona semigroup, search for feasibility conditions,
(4) constructions of new graph based stable groups and semigroups.

References

[1] B. Bollobas, *Extremal graph theory*, Academic Press, London, 1978.

[2] M. Polak, U. Romańczuk, V. Ustimenko and A. Wróblewska, "On the applications of Extremal Graph Theory to Coding Theory and Cryptography", *Electronic Notes in Discrete Mathema Discrete Mathematics*, N 43, 2013, p. 329-342. DOI: https://doi.org/10.1016/j.endm.2013.07.051

[3] F. Lazebnik, V. Ustimenko and A. J.Woldar, "A new series of dense graphs of high girth", *Bulletin of the American Mathematical Society* (N.S.) 32, no. 1, 1995, pp. 73-79

[4] V. Ustimenko, "On extremal graph theory and symbolic computations", *Dopovidi National Academy of Sciences of Ukraine*, , N2, 2013, pp. 42-49.

[5] V. Ustimenko, U. Romańczuk-Polubiec, A. Wróblewska, M. Polak and E. Zhupa, "On the implementation of new symmetric ciphers based on non-bijective multivariate maps", *Proceedings of the 2018 Federated Conference on Computer Science and Information Systems*, M.Ganzha, L. Maciaszek, M. Paprzycki (eds). ACSIS, Vol. 15, 2018, pp. 397-405 DOI: http://dx.doi.org/10.15439/2018F204

[6] V. Ustimenko, U. Romańczuk-Polubiec, A. Wróblewska, M. Polak and E. Zhupa, "On the constructions of new symmetric ciphers based on non-bijective multivariate maps of prescribed degree", *Security and Communication Networks*, Volume 2019, Article ID 2137561, 15 pages. DOI: https://doi.org/10.1155/2019/2137561

[7] M. Klisowski and V. Ustimenko, "Graph based cubical multivariate maps and their crypto-graphical applications", *Advances on Superelliptic curves and their Applications, IOS Press, NATO Science for Peace and Security series -D: Information and Communication Security*, vol 41, 2014, pp. 305 -327.

[8] U. Romańczuk-Polubiec and V. Ustimenko, "On Multivariate Cryptosystems Based on Polynomially Compressed Maps with Invertible Decompositions", *Cryptography and Security Systems, Third International Conference, CSS 2014, Lublin, Poland, September 22-24, 2014. Proceedings, Communications in Computer and Information Science*, 448, 2014, pp. 23-37. DOI: https://doi.org/10.1007/978-3-662-44893-9_3

[9] U. Romańczuk-Polubiec and V. Ustimenko, "On two windows multivariate cryptosystem depending on random parameters", *Algebra and Discrete Mathematics*, Volume 19, Number 1, 2015, pp. 101-129.

[10] U. Romańczuk and V. Ustimenko, "On Families of Graphs of Large Cycle Indicator, Matrices of Large Order and Key Exchange Protocols With Nonlinear Polynomial Maps of Small Degree", *Mathematics in Computer Science*, June 2012, Volume 6, Issue 2, pp 167-180, DOI: https://doi.org/10.1007/s11786-012-0115-8

[11] U. Romańczuk-Polubiec and V. Ustimenko, "On new key exchange multivariate protocols based on pseudorandom walks on incidence structures", *Dopovidi National Academy of Sciences of Ukraine*, No. 1, 2015, pp 41-49. DOI: https://doi.org/10.15407/dopovidi2015.01.041

[12] V. Ustimenko, "On algebraic graph theory and non-bijective maps in cryptography", *Algebra and Discrete Mathematics*, Volume 20, Number 1, 2015, pp. 152-170.

[13] V. Ustimenko, "Explicit constructions of extremal graphs and new multivariate cryptosystems", *Studia Scientarium Mathematicarium Hungarica* (Proceedings of Central. European Conference on Cryptology 2014, Budapest), vol 52, issue 2, June 2015, pp 185-204. DOI: https://doi.org/10.1556/012.2015.52.2.1312

[14] V. Ustimenko, "On the families of stable transformations of large order and their crypto-graphical applications", *Tatra Mt. Math. Publ.*, 70, 2017, pp. 107-117. DOI: https://doi.org/10.1515/tmmp-2017-0021

[15] V. Ustimenko, "On new symbolic key exchange protocols and cryptosystems based on hidden tame homomorphism", *Dopovidi National Academy of Sciences of Ukraine*, N. 10, 2018, pp. 26-36. DOI: https://doi.org/10.15407/dopovidi2018.10.026

[16] V. Ustimenko and M. Klisowski, "On Noncommutative Cryptography with cubical multiva-riate maps of predictable density", *Proceedings of "Computing 2019" conference, London, 16-17, July*, In: Arai K., Bhatia R., Kapoor S. (eds) Intelligent Computing. CompCom 2019. Advances in Intelligent Systems and Computing, vol 998. Springer, Cham. pp. 654-674. DOI: https://doi.org/10.1007/978-3-030-22868-2_47

[17] A. G. Myasnikov, V. Shpilrain and Alexander Ushakov, *Non-commutative Cryptography and Complexity of Group-theoretic Problems*, Mathematical Surveys and Monographs, American Mathematical Society, Volume 177, 2011. DOI: http://dx.doi.org/10.1090/surv/177

[18] P.L.K. Priyadarsini, "A Survey on some Applications of Graph Theory in Cryptography", *Journal of Discrete Mathematical Sciences and Cryptography*, 18:3, 2015, pp. 209-217. DOI: https://doi.org/10.1080/09720529.2013.878819

[19] V. Ustimenko, "On semigroups of multiplicative Cremona transformations and new solutions of Post Quantum Cryptography", *Cryptology ePrint Archive*, 133, 2019.

[20] O.S. Pustovit and V.O Ustimenko, "A new stream algorithms generating sensetive digests of digital documents", *Mathematical modelling in economics* (to appear).

[21] V. Ustimenko, "On desynchronised multivariate algorithms of El Gamal type for stable semigroups of affine Cremona group", *Theoretical and Applied Cybersecurity*, section: theoretical and cryptographic problems of cybersecurity , NTTU KPI, Kyiv, Vol. 1, No. 1, 2019, pp. 22-30.

[22] D. N. Moldovyan and N. A. Moldovyan, "A New Hard Problem over Non-commutative Finite Groups for Cryptographic Protocols", *International Conference on Mathematical Methods, Models, and Architectures for Computer Network Security*, MMM-ACNS 2010: Computer Network Security pp. 183-194. DOI: https://doi.org/10.1007/978-3-642-14706-7_14

[23] L. Sakalauskas and P. Tvarijonas, "A. Raulynaitis, Key Agreement Protocol (KAP) Using Conjugacy and Discrete Logarithm Problema in Group Representation Level", *INFORMATICA*, 2007, Vol. 18, No. 1, pp. 115-124

[24] V. Shpilrain and A. Ushakov, "The conjugacy search problem in public key cryptography: unnecessary and insufficient", *Applicable Algebra in Engineering, Communication and Computing*, August 2006, Volume 17, Issue 3-4, pp. 285-289. DOI: https://doi.org/10.1007/s00200-006-0009-6

[25] D. Kahrobaei and B. Khan, "A non-commutative generalization of ElGamal key exchange using polycyclic groups", *In IEEE GLOBECOM 2006 - 2006 Global Telecommunications Conference* [4150920]. DOI: https://doi.org/10.1109/GLOCOM.2006.290

[26] A. Myasnikov, V. Shpilrain and A. Ushakov, *Group-based Cryptography*, Advanced Courses in Mathematics - CRM Barcelona, Birkhäuser Basel, XV, p. 183, 2008. DOI: https://doi.org/10.1007/978-3-7643-8827-0

[27] Zhenfu Cao, "New Directions of Modern Cryptography", *Boca Raton: CRC Press, Taylor & Francis Group*, 2012, ISBN 978-1-4665-0140-9.

[28] B. Fine, et. al. "Aspects of Non abelian Group Based Cryptography: A Survey and Open Problems". *ArXiv*:1103.4093.

[29] I. Anshel, M. Anshel, D. Goldfeld, "An algebraic method for public-key cryptography", *Mathematical Research Letters*, 1999, 6(3-4), pp. 287-291.

[30] S. R. Blackburn and S. D. Galbraith, "Cryptanalysis of two cryptosystems based on group actions", In: Advances in Cryptology-ASIACRYPT '99. *Lecture Notes in Computer Science*, Springer, Berlin, 1999, vol. 1716, pp. 52-61.

[31] C. Ko, K.H., Lee, S.J., Cheon, J.H., Han, J.W., Kang, J.S. and Park, C., "New public-key cryptosystem using braid groups", In: Advances in Cryptology-CRYPTO 2000, Santa Barbara, CA. *Lecture Notes in Computer Science*, Springer, Berlin, 2000, vol. 1880, pp. 166-83. DOI: https://doi.org/10.1007/3-540-44598-6_10

[32] G. Maze, C. Monico, J. Rosenthal, "Public key cryptography based on semigroup actions", *Advances in Mathematics of Communications*, 2007, 1(4), pp. 489-507. DOI: https://doi.org/10.3934/amc.2007.1.489

[33] P. H. Kropholler, S.J. Pride , W.A.M. Othman K.B. Wong and P.C. Wong, "Properties of certain semigroups and their potential as platforms for cryptosystems", *Semigroup Forum*, 2010, Vol. 81, pp. 172-186. DOI: https://doi.org/10.1007/s00233-010-9248-8

[34] J. A. Lopez Ramos, J. Rosenthal, D. Schipani and R. Schnyder, "Group key management based on semigroup actions", Journal of Algebra and its applications, vol.16, No. 8, 2017. DOI: https://doi.org/10.1142/S0219498817501481

[35] G. Kumar and H. Saini, "Novel Noncommutative Cryptography Scheme Using Extra Special Group", Security and Communication Networks ,Volume 2017, Article ID 9036382, 21 pages. DOI: https://doi.org/10.1155/2017/9036382

[36] J. Ding., J. E. Gower and D. S. Schmidt, *Multivariate Public Key Cryptosystems*, Advances in Information Security, Springer, p. 260 v. 25, 2006. DOI: https://doi.org/10.1007/978-0-387-36946-4

[37] N. Koblitz, *Algebraic aspects of cryptography*, Springer, Berlin, Heidelberg, 1998. DOI: https://doi.org/10.1007/978-3-662-03642-6

[38] L. Goubin, J.Patarin, Bo-Yin Yang, *Multivariate Cryptography*, In: van Tilborg H.C.A., Jajodia S. (eds) Encyclopedia of Cryptography and Security. Springer, Boston, MA, (2nd Ed.), 2011, pp. 824-828. DOI: https://doi.org/10.1007/978-1-4419-5906-5

[39] R. Wagner, M. R. Magyarik, "A Public-Key Cryptosystem Based on the Word N Problem", *Advances in Cryptology*, Proceedings of CRYPTO

’84, Santa Barbara, California, USA, August 19-22, 1984. DOI: https://doi.org/10.1007/3-540-39568-7_3

[40] V. Ustimenko, “On new multivariate cryptosystems based on hidden Eulerian equations”, *Reports of Nath Acad of Sci, Ukraine*, 2017. No. 5, pp 17-24. DOI: https://doi.org/10.15407/dopovidi2017.05.017

[41] V. Ustimenko, “On new multivariate cryptosystems based on hidden Eulerian equations over finite fields”, *Cryptology ePrint Archive*, 093, 2017. 111

[42] V. Ustimenko and M. Klisowski, “On Noncommutative Cryptography and homomorphism of stable cubical multivariate transformation groups of infinite dimensional affine spaces”, *Cryptology ePrint Archive*, 593, 2019.

[43] V. Ustimenko, “Maximality of affine group, hidden graph cryptosystem and graph’s stream ciphers”, *Journal of Algebra and Discrete Mathematics*, 2004, v.10, pp. 51-65.

[44] V. Ustimenko, “Linguistic Dynamical Systems, Graphs of Large Girth and Cryptography”, *Journal of Mathematical Sciences*, Springer, Vol.140, N3, 2007, pp. 412-434. DOI: https://doi.org/10.1007/s10958-007-0453-2

[45] V. Ustimenko, “Graphs with Special Arcs and Cryptography”, *Acta Applicandae Mathematica*, November 2002, Volume 74, Issue 2, pp. 117-153 DOI: https://doi.org/10.1023/A:1020686216463

[46] A. Lubotzky, R. Phillips and P. Sarnak, “Ramanujan graphs”, *Combinatorica*, September 1988, Volume 8, Issue 3, pp. 261-277, DOI: https://doi.org/10.1007/BF02126799

7th Workshop on Advances in Programming Languages

PROGRAMMING languages are programmers' most basic tools. With appropriate programming languages one can drastically reduce the cost of building new applications as well as maintaining existing ones. In the last decades there have been many advances in programming languages technology in traditional programming paradigms such as functional, logic, and object-oriented programming, as well as the development of new paradigms such as aspect-oriented programming. The main driving force was and will be to better express programmers' ideas. Therefore, research in programming languages is an endless activity and the core of computer science. New language features, new programming paradigms, and better compile-time and run-time mechanisms can be foreseen in the future.

The aims of this event is to provide a forum for exchange of ideas and experience in topics concerned with programming languages and systems. Original papers and implementation reports are invited in all areas of programming languages.

TOPICS

Major topics of interest include but are not limited to the following:

- Automata theory and applications
- Compiling techniques
- Context-oriented programming languages to specify the behavior of software systems and dynamic adaptations
- Domain-specific languages
- Formal semantics and syntax
- Generative and generic programming
- Grammarware and grammar based systems
- Knowledge engineering languages, integration of knowledge engineering and software engineering
- Languages and tools for trustworthy computing
- Language theory and applications
- Language concepts, design and implementation
- Markup languages (XML)
- Metamodeling and modeling languages
- Model-driven engineering languages and systems
- Practical experiences with programming languages
- Program analysis, optimization and verification
- Programming paradigms (aspect-oriented, functional, logic, object-oriented, etc.)
- Proof theory for programs
- Type systems
- Virtual machines and just-in-time compilation
- Visual programming languages

STEERING COMMITTEE

- **Janousek, Jan,** Czech Technical University, Czech Republic
- **Luković, Ivan,** University of Novi Sad, Serbia
- **Mernik, Marjan,** University of Maribor, Slovenia
- **Slivnik, Boštjan,** University of Ljubljana, Slovenia

EVENT CHAIRS

- **Varanda Pereira, Maria João, Instituto Politecnico de Braganca,** Portugal

PROGRAM COMMITTEE

- **Barisic, Ankica,** Universidade Nova de Lisboa, Portugal
- **Fernandes, João Paulo,** Universidade de Coimbra
- **Horvath, Zoltan,** Eotvos Lorand University, Hungary
- **Janousek, Jan,** Czech Technical University, Czech Republic
- **Kardaş, Geylani,** Ege University International Computer Institute, Turkey
- **Kern, Heiko,** University of Leipzig, Germany
- **Kollár, Ján,** Technical University of Kosice, Slovakia
- **Kosar, Tomaž,** University of Maribor, Slovenia
- **Lopes Gançarski, Alda,** TELECOM SudParis, Evry, France
- **Luković, Ivan,** University of Novi Sad, Serbia
- **Mandreoli, Federica,** University of Modena, Italy
- **Martínez López, Pablo E. "Fidel",** Universidad Nacional de Quilmes, Argentina
- **Mernik, Marjan,** University of Maribor, Slovenia
- **Milašinović, Boris,** University of Zagreb Faculty of Electrical Engineering and Computing, Croatia
- **Pai, Rekha,** National Institute of Technology Calicut, India
- **Papaspyrou, Nikolaos,** National Technical University of Athens, Greece
- **Porubän, Jaroslav,** Technical University of Kosice, Slovakia
- **Rangel Henriques, Pedro,** Universidade do Minho, Portugal
- **Saraiva, João,** Universidade do Minho, Portugal
- **Sierra Rodríguez, José Luis,** Universidad Complutense de Madrid, Spain
- **Slivnik, Boštjan,** University of Ljubljana, Slovenia

Position Papers of the Federated Conference on Computer Science and Information Systems pp. 49–56
DOI: 10.15439/2019F355
ISSN 2300-5963 ACSIS, Vol. 19

A formal method to detect possible P4_{16} specific errors

Gabriella Tóth
Eötvös Loránd University
Faculty of Informatics
1/C. Pázmány Péter sny, Budapest, 1117, Hungary
Orcid Id: 0000-0001-9657-7231
Email: kistoth@inf.elte.hu

Máté Tejfel
Eötvös Loránd University
Faculty of Informatics
1/C. Pázmány Péter sny, Budapest, 1117, Hungary
Orcid Id: 0000-0001-8982-1398
Email: matej@inf.elte.hu

***Abstract*—P4 is a programming language to develop data processing of networks. This kind of programs are used in network devices – like switches – to describe the way of forwarding the received packets to the proper device. Checking the correctness of these programs is not an obvious task, because they can easily hide the run time errors. We are working on a method to detect violation of P4 specific properties. The method is based on a rule system, which can detect suspicious program parts and indicate the violated property. It helps to detect and correct real errors easily. As a first step, we introduce the main idea, dealing with the access of invalid header and uninitialized fields. We also present a case study to demonstrate the applicability of the method.**

I. Introduction

P4 [1] is a domain specific programming language to develop data plane of network devices. P4 makes it possible to develop target independent, protocol independent solutions for data plane processing. Budiu and Dodd [2] describe partially the main structure and the design goals of the most recent version of the language.

Although it opened a new dimension in the network data plane, it left the safety of bound protocols. Therefore, P4 developers need to be more prudent to create correct programs, or a proper solution need to be invented to detect different source of errors.

Errors are hardly detectable, because P4 easily hides them, and we can only recognize them from the bad or suspicious behavior. For example, if the program reads a field of an invalid header then it will get an unspecified value, with which it will continue the execution and it will not stop and sign the problem.

One solution can be accurate testing, but this can be too expensive and bounded. Formal methods can be more applicable. In the near past, different verification tools were published [3], [4], [5] to work on the correctness of P4 programs, with detecting specified properties, and give the opportunity for developers to correct them.

P4 has two main release version: P4_{14} [1], [6] and P4_{16} [2], [7]. There are main differences between them, for example in the structure, the syntax of the code and the deparsing phase. We work with the new one, and we would like to create firstly a method, secondly a tool to use them to check safety properties.

In this paper we introduce the first step forward a P4 specific, formal program property verification method. In this step we produce a rule system to detect different errors in the programs. Now, we work only with one possible error type, which can be caused simply by inattention and cannot be found easily. We define a property based on the validity of headers and fields, which says that, invalid headers or uninitialized fields should not be read or written, because it can cause undefined behavior. Section II shows some related works. In Section III we introduce some example for this problem. Section IV introduces the method specification and after it, we present its usage in a concrete case study in Section V.

A. Background on P4

Figure 1 contains a P4 program used by our case study. Main process of P4 programs is to get a bitstream as an input, extract the information of headers (with the parser) and modify them (with different actions) to create the new packet, which is sent forward to the network. There are some main structural units of the programs. Headers (lines 1-25) describe the handled header information about the packets. Parser (lines 27-48) extracts the data from the input bitstream to header instances. Control functions (lines 57-93) call the match-action tables, and handle the modification of headers with them. Match-action tables (lines 76-86) call actions. They work in a similar way as table lookup. They matches concrete fields of headers – named keys – with given values, and according to the result it executes an action call. The program only defines templates of the match-action tables, which contain a set of fields and a set of description of actions – their name and parameters. The concrete pairs of matching values and description

This work has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.3-VEKOP-16-2017-00002)

of action are coming from an external controller during run time. Actions define the modification of the values of headers. After the modification, the deparser builds the new packet from the created header data as an output bitstream and forwards it to the network.

Figure 1 shows an example, where there are two header type: $ethernet$ and $ipv4$. After the input packet arrives – as a bitstream – the parser will extract the header $ethernet$. If the value of field $ethernetType$ is $0x800$, then it will extract the header $ipv4$ too, otherwise is will not extract any other header. $MyIngress$ control function contains the modification part of the program. It has one table, which can call three different actions. One of the actions is the $ipv_forward$, which change the $srcAddr$, $dstAddr$ and $mathitttl$ fields of the $ipv4$. Another is the action $ipv4_new$, which tries to create a new $ipv4$, with the method $setValid$, and some assignments. The last one is the action $drop$, which only drops the processed packet. The main process of the control function is a branch, which will execute the table if the header $ipv4$ is valid. After the modification of the headers, the deparser will produce the output packet – as a bitstrem – which will contain both of the mentioned headers.

II. Related Work

In the near past different P4 specific verification tools were published. We would like to highlight four of them, which work with the previous main version of P4 – named P4_14.

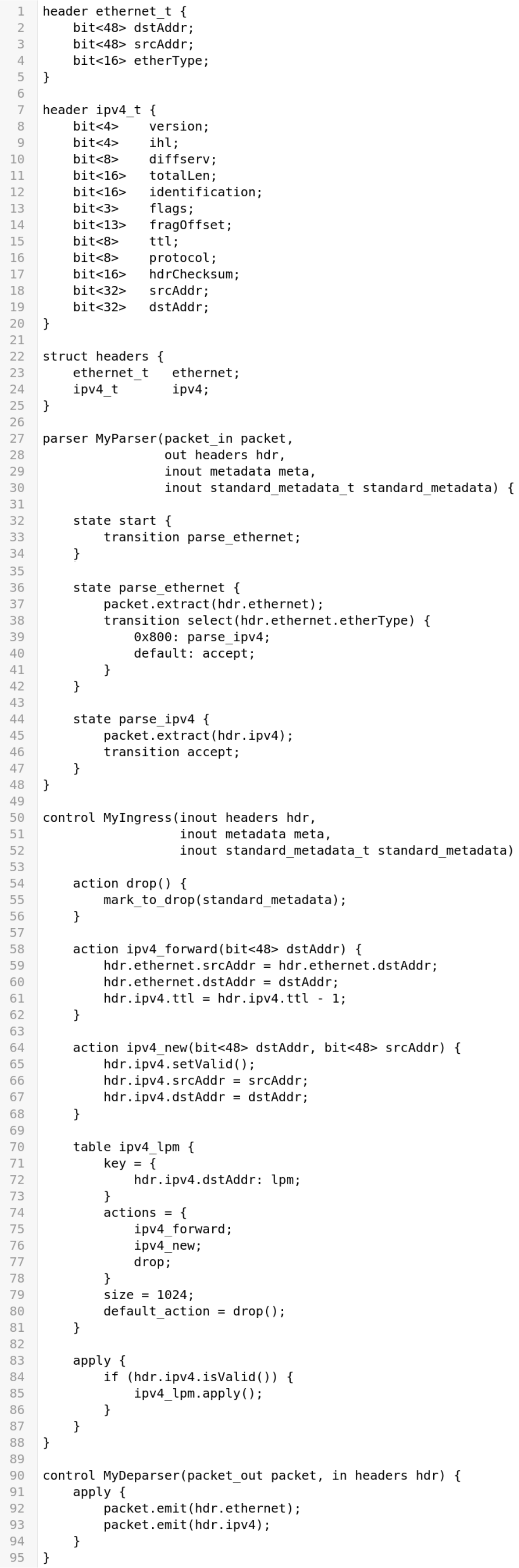
Assert-P4 [3] does not appoint specific properties to check, they entrust it to the developer, who can add assertions to the source code, and the tool will check their correctness. From the annotated program, it creates a C-model in which it examines the properties working based on symbolic execution – using Klee [8].

P4V [4] uses another approach of the problem. They transform the P4 code into GCL, create logical formulas from the GCL description, and check their satisfiability with Z3 [9].

Vera [5] uses also symbolic execution. It transforms the source code to SEFL, which is a modeling language to define state machines. The state machine represents every possible path of the execution, with symbolic values – using Symnet [10]. Vera works with predefined properties.

P4K [11] is a solution, which uses the $\mathbb{K}$ framework [12]. It presents the operational semantics of P4 and based on this semantics it can verify simple P4 properties using the reachability rule system of $\mathbb{K}$ [13].

All of the mentioned tools deal with validity checking, but neither of them mentioned the problem of uninitialized fields. Our solution extended the invalid header monitoring to the deparsing phase and to the usage of keys of tables. Our solution also checks the usage of uninitialized fields. Most of the proposed



```
header ethernet_t {
    bit<48> dstAddr;
    bit<48> srcAddr;
    bit<16> etherType;
}

header ipv4_t {
    bit<4>    version;
    bit<4>    ihl;
    bit<8>    diffserv;
    bit<16>   totalLen;
    bit<16>   identification;
    bit<3>    flags;
    bit<13>   fragOffset;
    bit<8>    ttl;
    bit<8>    protocol;
    bit<16>   hdrChecksum;
    bit<32>   srcAddr;
    bit<32>   dstAddr;
}

struct headers {
    ethernet_t   ethernet;
    ipv4_t       ipv4;
}

parser MyParser(packet_in packet,
                out headers hdr,
                inout metadata meta,
                inout standard_metadata_t standard_metadata) {

    state start {
        transition parse_ethernet;
    }

    state parse_ethernet {
        packet.extract(hdr.ethernet);
        transition select(hdr.ethernet.etherType) {
            0x800: parse_ipv4;
            default: accept;
        }
    }

    state parse_ipv4 {
        packet.extract(hdr.ipv4);
        transition accept;
    }
}

control MyIngress(inout headers hdr,
                  inout metadata meta,
                  inout standard_metadata_t standard_metadata)

    action drop() {
        mark_to_drop(standard_metadata);
    }

    action ipv4_forward(bit<48> dstAddr) {
        hdr.ethernet.srcAddr = hdr.ethernet.dstAddr;
        hdr.ethernet.dstAddr = dstAddr;
        hdr.ipv4.ttl = hdr.ipv4.ttl - 1;
    }

    action ipv4_new(bit<48> dstAddr, bit<48> srcAddr) {
        hdr.ipv4.setValid();
        hdr.ipv4.srcAddr = srcAddr;
        hdr.ipv4.dstAddr = dstAddr;
    }

    table ipv4_lpm {
        key = {
            hdr.ipv4.dstAddr: lpm;
        }
        actions = {
            ipv4_forward;
            ipv4_new;
            drop;
        }
        size = 1024;
        default_action = drop();
    }

    apply {
        if (hdr.ipv4.isValid()) {
            ipv4_lpm.apply();
        }
    }
}

control MyDeparser(packet_out packet, in headers hdr) {
    apply {
        packet.emit(hdr.ethernet);
        packet.emit(hdr.ipv4);
    }
}
```

Fig. 1. Example of P4 program

tools use some type of symbolic execution, but we try to stay in a formal solution, where we need no symbolic values. Although the current version of our method can not manage properly the problem of numerous different initial and final states, we are working on a solution to reduce this problem.

III. Motivation

The first property that we introduce is the validity of headers and their fields. Reading or writing of invalid headers or uninitialized fields can cause undefined behavior. Therefore, we would like to highlight the error prone usage of them.

There are two commands, which can be used to set the validity of headers. One of them is the $setValid()$ function, which can validate them. Calling of $setValid()$ has a side effect with which every field of the header become uninitialized. The other one is the $setInValid()$ function, which set the header to invalid, and all of its fields to uninitialized too. After the usage of these commands, the fields can be initialized explicitly with assignments.

Reading fields of invalid headers and uninitialized fields results unspecified values. Writing fields of invalid headers are unnecessary, because if we would like to use their values – for example in the output packet – we need to set the header to valid, which means that their fields become unspecified. Therefore, we need to avoid these type of codes, because it can easily lead to an error.

We would like to filter every occurrence of this problem, which can be in the control functions – for example in the conditions of branches, keys of tables and assignments. Using invalid header to emit in the deparsing phase is not an error, because P4 simply will not use it during the building of the packet, although it can be suspicious. Therefore, in this paper we will consider it as a possible error. Summary, in our rule system we would like to detect the usage of uninitialized fields and invalid headers as a key of a table, part of an assignment or during deparsing.

In the following sections we will link to the initialized fields as valid and the uninitialized fields as invalid fields, for simplification.

IV. Method of property checking

The method has two main parts. First is a preprocessing phase in which the initial states, final states and core program are produced. The initial states will be created from the source of parser, the final states will be created from the source of deparser, and the program will be produced from the control functions. The second phase is the usage of the rule system, which detects the errors. The calculation examines that, the execution can reach one of the final states from every initial state. This rule system is similar to an axiomatic semantics, but it has a more complex environment structure and additional side conditions.

$$S \in State :$$
$$Condition \times PacketInfo \times Environment$$

Fig. 2. Type of states

A. Preprocessing – Initial state

To use the method, first we need to preproccess the P4 code to produce the initial states. This first phase collects the used headers, and creates an empty environment, where everything is invalid. Than it analyses the parsing phase and calculates the different packet information and initial environments.

Parsing is a state machine with two final states – $accept$ and $reject$. Initial states will contain those paths, which start with the $start$ state and finish in the $accept$ state. They will use the conditions of branches to describe the different input packets in the packet information.

States will be represented with a triple – it is showed by Figure 2. The first element is the collected conditions, which is $True$ in the initial state, and is changed during the usage of the rule system. The second element is the packet information, which identifies the different input packets. The third part is the used environment.

$Condition$ collects the conditions of branches from the control functions as a conjunctive formula. There will be statements for validity checking of headers.

$PacketInfo$ and $Environment$ contains information about the input packet and the created headers. Suppose their is a parsing phase, which first extracts an $ethernet$ header, and after that it branch according to the value of $ethertype$ field. If its value is $0x800$ then it will extract $ipv4$, if its value is $0x86DD$ then it will extract $ipv6$ header, otherwise it will not extract any other headers. In this case, there can be three different initial packets, which can be described with the following formulas:

First case:

$$(ethernet.ethertype = 0x800;$$
$$ethernet = valid,\ ipv4 = valid,\ ipv6 = invalid)$$

Second case:

$$(ethernet.ethertype = 0x86DD;$$
$$ethernet = valid,\ ipv4 = invalid,\ ipv6 = valid)$$

Third case:

$$(ethernet.ethertype \neq 0x800$$
$$\wedge\ ethernet.ethertype \neq 0x86DD;$$
$$ethernet = valid,\ ipv4 = invalid,\ ipv6 = invalid)$$

Of course the environment description has concrete initial information about the fields of headers. It contains every headers and fields of the headers with their validity. In the parsing phase only total headers are parsed, therefore if a header is valid, then its fields will be valid too. The environment also contains an additional information, the *drop* flag, which shows the packet is set to drop or not.

In the environment, now we only collect the headers, their fields and their validity information, but as future work we plan to extend this information with concrete values to make possible the analysis of more precise properties.

B. Preprocessing – Final state

Deparsing phase is implemented as a control function in P4, so it can also contains branches. During deparsing the emit commands determine which headers and in which order are added to the output packet. If this header and its fields are valid then everything is fine. If the header is invalid, then the program will not add it to the packet. It is allowed to use it, but in our case, we would like to sign any suspicious case, therefore we will handle it as a possible error source. Emitting an uninitialized field of a valid header is also wrong, because it will use an unspecified value, so we would like to prevent it.

As mentioned above, the *drop* flag is part of the environment. Therefore, there will be a final state, which is always a possible one: which contains the *drop* variable with the value *1*, and every header and field can be valid or invalid. This state describes that case, when the packet is dropped. In every other case the *drop* flag needs to be *0*, and the value of headers and fields is defined.

The conditions ($Condition$ and $Packet$ $information$) of the final states are filled with $True$: $(True, (True, E)))$, because in the end of the calculation we will be able to restrict them – Rules 12 and 13 from Figure 3. There can be more final states, so there is a rule, with which the reachable one can be chosen – Rule 14.

C. Preprocessing – Core program

We need to create the core program for the rule system, so we unwrap the table applications and the action calls and concatenate the control functions – except for the deparser. We will get a sequentially program which contains every aspect of the code – for example the parameters of the actions – so we will be able to easily extend the property checking with new detection.

D. The rule system

During a deduction we prove that the program will reach one of the final state from all of the possible initial states. From the preprocessing phase we get one or more different initial states, and we need to verify the properties started from all of them. At the beginning of the method we can choose the conditions to $True$, because those will change during the code processing by adding other conditions of the branches with a conjunction.

The basic structure of the rule system is similar to the axiomatic semantics of P4. Although it is stricter than the basic behavior of the P4 programs and the used program states are extended with the above mentioned way. The rules will be inference rules in which the $PacketInfo$ part will never be changed, because it is used as an identification of the input packet – except for the case of Rule 13. Therefore, in the end of the calculation we will be able to tell more specified information about the possible errors.

Figure 3 shows the rule system of the main verification to detect errors. In the system, the S notations mean the statements, which have 3 main parts. C is a condition, P is a packet information and E is an environment. Therefore, every variants of these letters means the same type of element. On the right side of some rules there is a side condition, $S \vdash \{x\}$ – where x can be a condition of branch, a key of a table or one side of an assignment – statement. It checks the calculability of every element of the given set of statements by knowing the C conditions and E environment of the S state. Here calculability means that, every used field and header is valid.

In the rules, there are some specified expressions. Rule 1 and 2 describe the assignments. There can be two type of them: when we give value to a field (Rule 1) – in this case only the field become valid –, and when we give value to a header with a list (Rule 2) – here every field and the header are rewritten to valid. In the right side of the rule in the description of the environment there can be an expression (for example $\{S\ [E \mid\mid E[h.f \rightarrow valid]]\}$), which means that we use the S state with some modification in the E environment , especially the $h.f$ fields validity will be changed to valid. In the Rule 2, the $h.fields$ is used, which means, we use every field of header h, and in this case we set them to $valid$. In the right side of these rules, the checking of the used statements is appeared. It checks the validity of every part of the assignments, because the commands write the left side of the assignments and reads the right side.

Rules 3 and 4 process the setting of the validity of a header. The first rule sets it to valid, and its every field to invalid – because of the side effect. The other rule sets the given header and its fields to invalid.

During the process of the packet, there can be cases, when we need to drop the packet. Rule 5 describes this function, and simply set the *drop* flag to *1*.

The next three rules (Rules 6, 7 and 8) are the extended version of the common programming structures. First describes the sequence, second and third describe the branches with and without an else case. In the last two rules, it needs to check the validity of the condition of the branch.

In the last one, it uses $S_1 \wedge \neg b \Rightarrow S_2$, which can be rewritten to the following: $S_1.C \wedge toCond(S_1.E) \wedge \neg b \supset (S_2.C \wedge toCond(S_2.E))$, where $toCond(S.E)$

Let it be: $h \in \mathit{Headers},\ f \in \mathit{Fields}$

$$1.\quad \frac{}{\{S\}\ h.f = \mathit{exp}\ \{S\ [E\ ||\ E[h.f \to \mathit{valid}]]\}}\ \ S \vdash \{\mathit{exp},\ h,\ h.f\}$$

$$2.\quad \frac{}{\{S\}\ h = \mathit{list}\ \{S\ [E\ ||\ E[h \to \mathit{valid}, h.\mathit{fields} \to \mathit{valid}]]\}}\ \ S \vdash \mathit{list} \cup \{h\}$$

$$3.\quad \frac{}{\{S\}\ h.\mathit{setValid}()\ \{S\ [E\ ||\ E[h \to \mathit{valid}, h.\mathit{fields} \to \mathit{invalid}]]\}}$$

$$4.\quad \frac{}{\{S\}\ h.\mathit{setInValid}()\ \{S\ [\ E\ ||\ E[h \to \mathit{invalid}, h.\mathit{fields} \to \mathit{invalid}]]\}}$$

$$5.\quad \frac{}{\{S\}\ \mathit{mark\_to\_drop}(..)\ \{S\ [E\ ||\ E[\mathit{drop} \to 1]]\}}$$

$$6.\quad \frac{\{S_1\}\ Pr_1\ \{S_2\} \qquad \{S_2\}\ Pr_2\ \{S_3\}}{\{S_1\}\ Pr_1; Pr_2\ \{S_3\}}$$

$$7.\quad \frac{\{S_1\ [C\ ||\ C \wedge b]\}\ Pr_1\ \{S_2\} \qquad \{S_1\ [C\ ||\ C \wedge \neg b]\}\ Pr_2\ \{S_2\}}{\{S_1\}\ \mathit{if}\ (b)\ \{Pr_1\}\ \mathit{else}\ \{Pr_2\}\ \{S_2\}}\ \ S_1 \vdash \{b\}$$

$$8.\quad \frac{\{S_1\ [C\ ||\ C \wedge b]\}\ Pr\ \{S_2\} \qquad S_1 \wedge \neg b \Rightarrow S_2}{\{S_1\}\ \mathit{if}\ (b)\ \{Pr\}\ \{S_2\}}\ \ S_1 \vdash \{b\}$$

$$9.\quad \frac{\{S_1\}\ A_1.\mathit{body}\ \{S_2\} \qquad \ldots \qquad \{S_1\}\ A_n.\mathit{body}\ \{S_2\} \qquad S_1 \Rightarrow S_2}{\{S_1\}\ \mathit{table}\ \mathit{keys} : K\ \mathit{actions} : \{A_1, ..., A_n\}\ \{S_2\}}\ \ S_1 \vdash K$$

$$10.\quad \frac{\{S_1\}\ A_1.\mathit{body}\ \{S_2\} \qquad \ldots \qquad \{S_1\}\ A_n.\mathit{body}\ \{S_2\} \qquad \{S_1\}\ A_{n+1}.\mathit{body}\ \{S_2\}}{\{S_1\}\ \mathit{table}\ \mathit{keys} : \{K\}\ \mathit{actions} : \{A_1, ..., A_n, A_{n+1}\}\ \{S_2\}}\ \ S_1 \vdash K$$

$$11.\quad \frac{S_1.C \supset C' \qquad \{S_1\ [C\ ||\ C']\}\ S\ \{S_2\}}{\{S_1\}\ Pr\ \{S_2\}} \qquad 12.\quad \frac{\{S_1\}\ Pr\ \{S_2\ [C\ ||\ C']\} \qquad C' \supset S_2.C}{\{S_1\}\ Pr\ \{S_2\}}$$

$$13.\quad \frac{\{S_1\}\ Pr\ \{S_2\ [P\ ||\ P_3]\} \qquad P_3 \supset S_2.P}{\{S_1\}\ Pr\ \{S_2\}}$$

$$14.\quad \frac{\{S_1\}\ Pr\ \{S_i\} \qquad i \in [2..n]}{\{S_1\}\ Pr\ \{S_2, \ldots S_n\}} \qquad 15.\quad \frac{\{S_1\}\ Pr\ \{S_m\} \qquad \ldots \qquad \{S_n\}\ Pr\ \{S_m\}}{\{S_1, \ldots, S_n\}\ Pr\ \{S_m\}}$$

Fig. 3. Rule system to verify validity

means the conversion of the environment to conditions. For example, if there is a header h, with value valid, it will create a condition like $h == \mathit{valid}$. So the expression checks that, the information of S_1 and the negated b implies the information of S_2.

The following two rules (Rules 9 and 10) describe the behavior of match-action tables, as a branch with n cases. There is validity checking of the keys in both rules. The previous one works with the tables without a default action, and the latter one works with default actions too.

The last five rules can be used to refine conditions and packet information and states. The first is the strengthen of the left side condition, the second is the weaken of the right side condition. The third one weakens the packet conditions. It is necessary during the deduction to strengthen the packet information of the final state, because at the beginning, we use True to describe it, and in the end it needs to match with the concrete description. The forth one say that if their is a deduction to an S_i state then there will be a deduction to more states, where one of them is S_i. In the deduction it means that we choose one final state. The last one can be used to separate the deduction based on the initial states.

E. Verification

The initial states will be created in preprocessing phase, which will separate the possible input packets. The final states and the core program will be produced too. We need to create a deduction from each of the initial states to one of the final states by using the inference rules. If there is any problem, the deduction will stop and we will be able to see the errors from the stuck paths. There are branches in the deduction tree. Beside a deduction has one stucked path, it can detect other errors, or it can work well in other paths too. So we can detect different errors in one proof, and determine the problems from the calculation of the incorrect conditions.

Producing a proof for every initial state, will mean that the given program is well defined, and it does not violent the examined program properties.

V. Case study

This section focuses to the P4 example, which is illustrated by Figure 1. We show the results of the verification and its phases.

A. Preprocessing – initial states

This phase of the preprocessing use the headers and the parsing phase. The environment of the initial states contains every headers and fields. There is an added flag – named *drop* – which represents the intention of drop the input packet. The parsing phase extracts the *ethernet* header and than the execution is branched. If condition *ethernet.ethertype* = *0x800* holds, it will extract an *ipv4* header, on other case it will not extract other headers. After the calculation of preprocessing, two possible initial states will be produced.

The produced two possible initial states are the followings:

```
I1 = (True, ethernet.ethertype = 0x800,
      [drop = 0,
       ethernet: ( valid, {
                   dstAddr: valid,
                   srcAddr: valid,
                   ethertype: valid}),
       ipv4:     ( valid, {
                   version: valid,
                   ...
                   dstAddr: valid})])
```

Fig. 4. First initial state

B. Preprocessing – final states

The final states calculation uses the code of the deparsing phase. In the example, there is a really simple deparser, which firstly emits the *ethernet*, and secondly emits the *ipv4* header. Therefore there will be two reachable environments. One of them describes the case, when the packet will be dropped, and the other fields and headers validity is not important – Figure 6 represents it with the marking *others* = ∗. The other describes the execution when the packet is not dropped, and the *ethernet*, *ipv4* and all of their fields are valid, because we would like to forward them.

```
I2 = (True, ethernet.ethertype ≠ 0x800,
      [drop = 0,
       ethernet: ( valid, {
                   dstAddr: valid,
                   srcAddr: valid,
                   ethertype: valid}),
       ipv4:     ( invalid, {
                   version: invalid,
                   ...
                   dstAddr: invalid})])
```

Fig. 5. Second initial state

```
F = {(True, True, [drop = 1, others = *]),
     (True, True,
      [drop = 0,
       ethernet: (valid, {
                  dstAddr: valid,
                  srcAddr: valid,
                  ethertype: valid}),
       ipv4: (valid, {
              version: valid,
              ...
              dstAddr: valid})])}
```

Fig. 6. The calculated final state

C. Preprocessing – executable program

This phase only use the code of the control functions. It concatenates the main code of the control functions, and unwrap the calls of the tables and actions.

```
Pr =
if (hdr.ipv4.isValid()) {
  table
    keys: {hdr.ipv4.dstAddr}
    actions: {
      ipv4_forward(bit<48> dstAddr) {
        hdr.ethernet.srcAddr = hdr.ethernet.dstAddr;
        hdr.ethernet.dstAddr = dstAddr;
        hdr.ipv4.ttl = hdr.ipv4.ttl - 1;
      },
      drop(){
        mark_to_drop(standard_metadata);
      },
      ipv4_new(bit<32> dstAddr, bit<32> srcAddr) {
        hdr.ipv4.setValid();
        hdr.ipv4.srcAddr = srcAddr;
        hdr.ipv4.dstAddr = dstAddr;
      }}
}
```

Fig. 7. Produced core program

$$\dfrac{\dfrac{\begin{array}{c}\dfrac{\dfrac{\checkmark}{\{I_{11}\}\ \mathit{hdr.ipv4.ttl} = ..\ \{I_{11}\}}}{\{I_{11}\}\ \mathit{hdr.ipv4.ttl} = ..\ \{F_2\}}\\ \vdots \\ \{I_{11}\}\ \mathit{hdr.ipv4.srcAddr} = ..;..\ \{F\}\end{array}\quad \dfrac{\dfrac{\dfrac{\checkmark}{\{I_{11}\}\ \mathit{mark\_to\_drop}(..)\ \{F_{11}\}}}{\{I_{11}\}\ \mathit{mark\_to\_drop}(..)\ \{F_1\}}}{\{I_{11}\}\ \mathit{mark\_to\_drop}(..)\ \{F\}}\quad \begin{array}{c}\dfrac{\lightning}{\{I_{12}\}\ \mathit{hdr.ipv4.dstAddr} = ..\ \{F_2\}}\\ \vdots \\ \{I_{11}\}\ \mathit{hdr.ipv4.setValid}();..\ \{F\}\end{array}}{\{I_{11}\}\ \mathit{tablekey}\ :\{..\}\ \mathit{actions}\ :\ \{\mathit{ipv4\_forward}(..)..,\ \mathit{drop}(..)\{..\},\ \mathit{ipv4\_new}(..)..\}\}\{F\}}\quad \dfrac{\checkmark}{(I_1 \wedge \neg b \supset F)}}{\{I_1\}\ \mathit{if}(\mathit{hdr.ipv4.isValid}())\{...\}\ \{F\}}$$

$I_{11} = (\mathit{hdr.ipv4.isValid}(), (\mathit{ethernet.ethertype} = 0x800, \{\mathit{drop} = 0, \mathit{ethernet} = \mathit{valid}, \mathit{ethernet.all} = \mathit{valid}, \mathit{ipv4} = \mathit{valid}, \mathit{ipv4.all} = \mathit{valid}\}))$

$I_{12} = (\mathit{True}, (\mathit{ethernet.ethertype} = 0x800, \{\mathit{drop} = 0, \mathit{thernet} = \mathit{valid}, \mathit{ethernet.all} = \mathit{valid}, \mathit{ipv4} = \mathit{valid}, \mathit{ipv4.srcAddr} = \mathit{valid}, \mathit{ipv4.others} = \mathit{invalid}\}))$

$F_{11} = (\mathit{ipv4.isValid}(), \mathit{ethernet.ethertype} = 0x800, [\mathit{drop} = 1, \mathit{others} = *])$

Fig. 8. Deduction of I_1

D. Using the system

Figure 8 contains the shorter version of the deduction from the first initial state. In the figure there are 4 main paths in the deduction. We took a short cut for the deduction. In parts where there are dots, there are branches, – because of the sequences – we just illustrate the last branch of all paths.

First path shows the effect of *ipv4_forward* action. It does not change the validity of the fields, so in the end it reaches the second version of the final state – the one in which the packet is not dropped.

The second path only contains the *mark_to_drop(..)* statement, which set the *drop* variable of the environment to *1*, therefore it reaches the first version of the final state – which describes the packet dropping.

The third path does not reach the given final states. According to the dropping, it should reach the second version of the final state, but it is not proper because of the validity of fields of the *ipv4* header. The *ipv4_new* action sets the validity of the *ipv4* header to valid, but its side effect also sets all of its fields to invalid. After that, it only reinitializes the *dstAddr* and the *srcAddr*, but there are others, which stay invalid, and there is no reachable final state for this environment.

The fourth path contains the execution, when the condition of the if statement was false. It is written with a logical formula, which is an implication, which has a *false* expression in the left side, so the result of the whole formula is *true*.

$$\dfrac{\dfrac{\checkmark}{\{I_{22}\}\ \mathit{table..}\ \{F\}}\quad \dfrac{\lightning}{I_2 \wedge \neg \mathit{ipv4.isValid}() \supset F}}{\{I_2\}\ \mathit{if}(\mathit{hdr.ipv4.isValid}())\{...\}\ \{F\}}$$

$I_{22} = (hdr.ipv4.isValid(), ethernet = valid, ethernet.all = valid, ipv4 = invalid, ipv4.all = invalid)$

Fig. 9. Deduction of I_2

Figure 9 produces the deduction from the second initial state. This deduction has one right and one wrong branch. The first one has no error, because of the condition – $\mathit{ipv4.isValid}()$ – and the environment – where $\mathit{ipv4} = \mathit{invalid}$ – are contradict each other. These type of paths can be accepted. This type of paths describes parts of the execution, which never runs.

The second path examines the case when the condition of the branch is false – so $\neg\mathit{ipv4.isValid}()$. In this case the left side of the implication is *true*, but it does not implicate any of the final states, because they do not have matching environments. It could match only with the final state, where the *drop* field is *0*, but there is no match between the environments. This result comes from the *ipv4* header, because in the initial state it is not valid, but in the possible reachable state it is valid. Therefore, the whole formula is false. In this case, one of the final states should be reachable from the initial state, because the executed program is a *skip* one,but this condition is not right.

VI. Future Work

A. More detailed method

First of all we will define the details of the method, for example the processing of the parser and deparser.

We will extend the method with other P4 specific properties. To work with other properties, we will need to follow the changes of fields with an environment, which will contain the values and other information – according to the checked properties – about the headers and their fields.

We will fix the disadvantage of the current method, it can not calculate all possible errors in one execution cycle. It stops in a path, when a problem is detected, and it does not continue the calculation. For example, if we use the Rule 9 in Figure 3, and the keys are not valid then we will not be able to detect the problems in the actions too, only if the first one is corrected. In the future we would like to figure out the solution to continue the calculation, for example with some supposed assertions.

The current solution can manage only simplified structure elements of P4 programs, therefore we will need to extend the rule system, and the statements with other P4 elements. For example now we do not work with metadata headers, constants.

B. Implementation

In the near future, we will implement the introduced method. Currently we are working on representing the rule system in Coq [14]. Later we will create a tool, which will be able to analyze an arbitrary P4 code according to our method, and sign the possible error source.

C. Applicability in complex examples

The final step is to be able to detect errors in larger, more complex programs too. We would like to extract the method with the possibility of the compositional processing, therefore we could simplify the deductions. We will extend the rule system with new rules, which will allow to process the core program with divided parts.

VII. Conclusions

This paper introduces a formal method for property checking of P4 programs. For the time being, it only works with the validity of the headers and the initialization of the fields, but it will be also complemented with other, P4 specific properties. The produced method have two main parts. The first one is the preprocessing, which prepares the information for the concrete calculations – that creates the initial and final states with collecting and merging the contents of core program. The second part is the deduction, which examines that, from every initial states the program can reach one of the final states. If we can prove it with the rules then we have a well defined P4 program. If there is any problem during the deduction then we will be able to denote the error from the valuated condition or the states.

References

[1] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 3, pp. 87–95, 2014. doi: 10.1145/2656877.2656890. [Online]. Available: http://dx.doi.org/10.1145/2656877.2656890

[2] M. Budiu and C. Dodd, "The p416 programming language," *SIGOPS Oper. Syst. Rev.*, vol. 51, no. 1, pp. 5–14, Sep. 2017. doi: 10.1145/3139645.3139648. [Online]. Available: http://dx.doi.org/10.1145/3139645.3139648

[3] L. Freire, M. Neves, L. Leal, K. Levchenko, A. Schaeffer-Filho, and M. Barcellos, "Uncovering Bugs in P4 Programs with Assertion-based Verification," in *Proceedings of the Symposium on SDN Research*, ser. SOSR '18. New York, NY, USA: ACM, 2018. doi: 10.1145/3185467.3185499. ISBN 978-1-4503-5664-0 pp. 4:1–4:7. [Online]. Available: http://dx.doi.org/10.1145/3185467.3185499

[4] J. Liu, W. Hallahan, C. Schlesinger, M. Sharif, J. Lee, R. Soulé, H. Wang, C. Caşcaval, N. McKeown, and N. Foster, "P4v: Practical Verification for Programmable Data Planes," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '18. New York, NY, USA: ACM, 2018. doi: 10.1145/3230543.3230582. ISBN 978-1-4503-5567-4 pp. 490–503. [Online]. Available: http://dx.doi.org/10.1145/3230543.3230582

[5] R. Stoenescu, D. Dumitrescu, M. Popovici, L. Negreanu, and C. Raiciu, "Debugging P4 Programs with Vera," in *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '18. New York, NY, USA: ACM, 2018. ISBN 978-1-4503-5567-4 pp. 518–532. [Online]. Available: http://dx.doi.org/10.1145/3230543.3230548

[6] (2018) The P4 Language Specification. [Online]. Available: https://p4.org/p4-spec/p4-14/v1.0.5/tex/p4.pdf

[7] (2018) $P4_{16}$ Language Specification. [Online]. Available: https://p4.org/p4-spec/docs/P4-16-v1.1.0-spec.pdf

[8] C. Cadar, D. Dunbar, and D. Engler, "Klee: Unassisted and Automatic Generation of High-coverage Tests for Complex Systems Programs," in *Proceedings of the 8th USENIX Conference on Operating Systems Design and Implementation*, ser. OSDI'08. Berkeley, CA, USA: USENIX Association, 2008, pp. 209–224. [Online]. Available: http://dl.acm.org/citation.cfm?id=1855741.1855756

[9] L. De Moura and N. Bjørner, "Z3: An Efficient SMT Solver," in *Proceedings of the Theory and Practice of Software, 14th International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, ser. TACAS'08/ETAPS'08. Berlin, Heidelberg: Springer-Verlag, 2008. doi: 10.1007/978-3-540-78800-3-24. ISBN 3-540-78799-2, 978-3-540-78799-0 pp. 337–340. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-78800-3-24

[10] R. Stoenescu, M. Popovici, L. Negreanu, and C. Raiciu, "Symnet: Scalable Symbolic Execution for Modern Networks," in *Proceedings of the 2016 ACM SIGCOMM Conference*, ser. SIGCOMM '16. New York, NY, USA: ACM, 2016. doi: 10.1145/2934872.2934881. ISBN 978-1-4503-4193-6 pp. 314–327. [Online]. Available: http://dx.doi.org/10.1145/2934872.2934881

[11] A. Kheradmand and G. Rosu, "P4K: A formal semantics of P4 and applications," *CoRR*, vol. abs/1804.01468, 2018. [Online]. Available: http://arxiv.org/abs/1804.01468

[12] G. Roşu, $\mathbb{K}$*: A semantic framework for programming languages and formal analysis tools*, 01 2017, pp. 186–206. [Online]. Available: http://dx.doi.org/10.3233/978-1-61499-810-5-186

[13] A. Stefănescu, D. Park, S. Yuwen, Y. Li, and G. Roşu, "Semantics-based Program Verifiers for All Languages," *SIGPLAN Not.*, vol. 51, no. 10, pp. 74–91, Oct. 2016. doi: 10.1145/3022671.2984027. [Online]. Available: http://dx.doi.org/10.1145/3022671.2984027

[14] The Reference Manual of the Coq. [Online]. Available: https://coq.inria.fr/distrib/current/refman/

4$^{\text{th}}$ International Workshop on Language Technologies and Applications

DEVELOPMENT of new technologies and various intelligent systems creates new possibilities for information processing. Natural Language Processing (NLP) addresses problems of automated understanding, processing, evaluation and generation of natural human languages. LTA workshop provides a venue for discussion and presenting innovative research in NLP domain, but not restricted, to: computational and mathematical modeling, analysis and processing of any forms (spoken, handwritten or text) of human language, interactions via Virtual Reality and Augmented Reality, Computational Intelligence models and applications but also other various applications in decision support systems. We welcome papers covering innovative applications and practical usage of theoretical aspects. The LTA workshop will provide an opportunity for researchers and professionals to discuss present and future challenges as well as potential collaboration for future progress in the field.

TOPICS

The submitted papers shall cover research and developments in all NLP aspects, such as (however this list is not exhaustive):

- Computational Intelligence methods applied to language & text processing
- text analysis
- language networks
- text classification
- language networks, resources and corpora
- document clustering
- various forms of text recognition
- machine translation
- intelligent text-to-speech (TTS) and speech-to-text (STT) methods
- authorship identification and verification
- author profiling
- plagiarism detection
- sentiment analysis
- NLP applications in education
- knowledge extraction and retrieval from text and natural language structures
- multi-modal and natural language interfaces
- innovative language-oriented applications and tools
- interactions models and applications via Virtual Reality and Augmented Reality
- NLP for text analysis in forensic linguistics and cybersecurity

EVENT CHAIRS

- **Damasevicius, Robertas,** Kaunas University of Technology, Lithuania
- **Martinčić – Ipšić, Sanda,** University of Rijeka, Croatia
- **Napoli, Christian,** Department of Mathematics and Informatics, University of Catania, Italy
- **Sanada, Haruko,** Rissho University, Japan
- **Woźniak, Marcin,** Institute of Mathematics, Silesian University of Technology, Poland

PROGRAM COMMITTEE

- **Artiemjew, Piotr,** University of Warmia and Mazury, Poland
- **Bajović, Dragana,** University of Novi Sad, Serbia
- **Burdescu, Dumitru Dan,** University of Craiova, Romania
- **Čukić, Bojan,** UNC Charlotte, United States
- **Dobrišek, Simon,** University of Ljubljana, Slovenia
- **Gelbukh, Alexander,** Instituto Politécnico Nacional, Mexico
- **Harbusch, Karin,** Universität Koblenz-Landau, Germany
- **Ivanović, Dragan,** University of Novi Sad, Serbia
- **Kapočiūtė-Dzikienė, Jurgita,** Vytautas Magnus University, Lithuania
- **Krilavičius, Tomas,** Vytautas Magnus University, Lithuania
- **Kurasova, Olga,** Vilnius University, Institute of Mathematics and Informatics, Lithuania
- **Lopata, Audrius,** Vilnius University, Lithuania
- **Madjarov, Gjorgji,** Ss. Cyril and Methodius University in Skopje, Faculty of Computer Science and Engineering, Macedonia
- **Marszałek, Zbigniew,** Silesian University of Technology, Poland
- **Maskeliūnas, Rytis,** Kaunas University of Technology, Lithuania
- **Matson, Eric T.,** Purdue University, United States
- **Meštrović, Ana,** University of Rijeka, Croatia
- **Mikelić-Preradović, Nives,** University of Zagreb, Croatia
- **Nowicki, Robert,** Czestochowa University of Technology, Poland
- **Połap, Dawid,** Institute of Mathematics, Silesian University of Technology, Poland
- **Stanković, Ranka,** University of Belgrade, Serbia
- **Starczewski, Janusz,** Czestochowa University of Technology, Poland

- **Szymański, Julian,** Gdansk University of Technology, Poland
- **Tahmasebi, Nina,** University of Gothenburg, Sweden
- **Tambouratzis, George,** Institute for Language and Speech Processing, Athena Research Centre, Greece
- **Trivodaliev, Kire,** Ss. Cyril and Methodius University in Skopje, Faculty of Computer Science and Engineering, Macedonia
- **Wang, Lipo,** Nanyang Technological University, Singapore
- **Wei, Wei,** School of Computer Science and EngineeringXi'an University of Technology, China

Position Papers of the Federated Conference on
Computer Science and Information Systems pp. 59–66

DOI: 10.15439/2019F254
ISSN 2300-5963 ACSIS, Vol. 19

Delta Analyzer: Tool-based Evaluation of Modified Requirements for an Efficient Development Effort Estimation in the RFQ Process

Konstantin Zichler
Advanced Engineering Projects
HELLA GmbH & Co. KGaA
Lippstadt, Germany
Email: konstantin.zichler@hella.com

Felix Ritter, Aaron Schul
Department of Computer Science
University of Applied Sciences and
Arts, Dortmund, Germany
Email: {felix.ritter001, aaron.schul002}
@stud.fh-dortmund.de

Steffen Helke
Department of Electrical Engineering
and Information Technology
South Westphalia University of Applied
Sciences, Hagen, Germany
Email: helke.steffen@fh-swf.de

***Abstract*—Once an automotive OEM decides to source a new component, a Request for Quotation (RFQ) is send to potential suppliers. Among other documents the RFQ contains a Component Requirements Specification (CRS), which describes the properties of the desired component. As a next step, the supplier has to evaluate the requirements and other boundary conditions of the RFQ and to provide an offer to the OEM. In case the supplier already developed a similar component in the past, it is possible to compare the CRS of the predecessor product with the actual CRS, to estimate the additional development effort. This activity is known as the delta analysis. Since no sufficient tool support is offered, this activity is still a predominantly manual task. The main challenge arises from the fact, that specification documents within the RFQ are provided in different office formats, written by different authors and therefore cannot be compared automatically with the CRS from the predecessor product. In our previous work, we presented the Requirements to Boilerplates Converter (R2BC), which automatically converts random natural language requirements into a predefined syntax. The aim of the approach is to facilitate a subsequent tool-based delta analysis. Consequently, we hereby introduce our proprietary developed Delta Analyzer (DA). This tool is based on Natural Language Processing (NLP) and allows to compare automatically two random specification documents. Moreover, the DA prioritizes requirements deltas according to their impact on development effort. As an output of the DA requirements engineers receive a delta report, which outlines the major differences between the requirements of the two CRS. We validate our approach by experiments on real-life specification documents.**

Keywords: Requirements engineering, delta analysis, natural language processing, requirements delta prioritization

I. Introduction

SUPPLIER development projects are often triggered by a Request for Quotation (RFQ) of an OEM. Within an RFQ, OEMs provide information about a desired component, which shall be delivered by the supplier. Together with project related information like target vehicles, Start of Production (SOP) dates and product volumes, each RFQ contain a Component Requirements Specification (CRS). This specification describes all requirements for the desired component. Based on this document the supplier project team evaluates during the RFQ phase, whether to quote for the RFQ or not. A schematic illustration of a typical RFQ process is depicted in Fig. 1. According to this process, the project team analyses the CRS of the OEM. The target of this evaluation is to estimate the effort, which is necessary to develop the component. To this end, a delta analysis is performed. Such analysis is the activity of comparing two requirements specifications to determine the differences, namely the deltas, between the listed requirements. This procedure is conducted, in case a successor of an already available product is to be developed and the requirements specifications of both the successor and the predecessor products are available. Because the effort to fulfill the requirements of the predecessor component is already known, the project team now only has to estimate the additional effort to fulfill the requirements of the successor CRS [1]. In the next step the effort estimation of the project team is used to provide an offer for the requested product to the OEM.

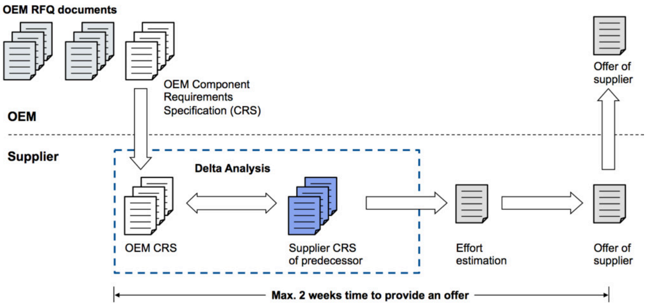

OEM RFQ documents
OEM Component Requirements Specification (CRS)
Offer of supplier
OEM
Supplier
Delta Analysis
OEM CRS
Supplier CRS of predecessor
Effort estimation
Offer of supplier
Max. 2 weeks time to provide an offer

Fig. 1. Schematic Illustration of the RFQ Process

Today, project teams are required to finish the RFQ phase within two weeks or even faster. To increase the efficiency of this process, project teams focus on the top ten requirements of a product. The top ten requirements describe the major characteristics of a product and have therefore a high impact on the development effort. The prioritization of these requirements and concentration on the evaluation of them, gives the project team the chance to quickly answer the question, whether the predecessor component is capable to fulfill the OEM requirements at all or a totally new development is necessary. Through the years, technical experts gathered knowledge on the effort necessary to adapt a given component according to changed customer requirements. By reading a CRS, experts are able to estimate whether a change in a parameter requires a new subcomponent and consequently the rework of the whole system. If for example, a given actuator with the specific characteristic of 5 Nm torque is required by the new OEM CRS to provide 10 Nm, experts can deduce that the fulfillment of this requirement would require a new motor. Since the motor is a major component of the actuator, further components like the gear may have to be adapted, as well. Following this type of consideration, technical experts can estimate the effort necessary to develop the changed component.

The delta analysis is still a predominantly manual task, which requires a lot of resources. Technical experts of the project team read up to several hundreds of pages and compare the predecessor CRS with the successor CRS. This work is tedious and time consuming. Currently, no sufficient tool support exists, that could support the project teams properly. Common tools, which are used by the industry allow the delta analysis for two states of the same document. To this end requirements, attributes and other information are stored as objects in a database. All changes made to these objects can be made visible by the database management system. The presented situation in the RFQ phase requires further functionality from a tool support. Time and again new requirements specifications are submitted to the supplier. These documents are written by various unknown authors and are provided in common office formats (e.g., PDF). Within the delta analysis these documents must be compared to the supplier specification, which was also written by another author. Therefore, a tool support is required, which can determine the requirements deltas between different documents.

In this work, we present our concept for an automated delta analysis and the architecture of the corresponding tool – the Delta Analyzer (DA). This tool allows the comparison of two completely different requirements documents. It uses Natural Language Processing (NLP) techniques, which is the basis of its flexibility. Also, we use our proprietary developed tool R2BC, which we presented in our previous work [1] as a prerequisite for the automated delta analysis. Based on NLP the R2BC converts random natural language requirements into predefined sentence structures, called boilerplates. Moreover, our concept for an automated delta analysis includes an algorithm for the prioritization of requirements deltas. This function prioritizes requirements deltas in the delta report in accordance to their impact on the development effort. All in all, our approach aims to decrease the amount of work needed during the RFQ phase.

The remainder of this work is structured as follows. Chapter II gives an overview of NLP techniques, which we apply in our approach. In Chapter III we present our concept for the automated delta analysis and the methodology for the prioritization of requirements deltas. We present the result of preliminary experiments with the DA in Chapter IV. In Chapter V, we give an overview of related work. Chapter VI summarizes the presented work and gives an outlook on our next research activities.

II. Fundamentals

A. NLP

Natural language processing is to be understood as the automated or semi-automated processing of natural language with the help of a computer. NLP connects various scientific fields. These are mainly linguistics and computer science, but there are also many links to psychology, philosophy, mathematics and logic [2].

The goal of NLP is usually the extraction of information out of a text through precise text analysis. This information is attached directly to the text via so-called annotations, which can later be used to define properties in program objects. By combining linguistics and computer science, NLP generally faces the challenge of having to solve both inherent problems of language and problems of automation. Linguistic issues include aspects such as ambiguity, sarcasm and slang or sayings. These can often not be explained solely by the given text, but are based on context and situation. In the implementation, it can then be hard to find consistent regularities as a basis for automation despite the linguistic features. But, if these challenges are understood correctly and are solved accordingly, NLP allows efficient text processing and information extraction, which can be used in a variety of applications, such as component requirements analyses.

B. GATE

One of the most commonly used tools for creating NLP applications is the *General Architecture for Text Engineering* (GATE). It allows for documents to be imported and then processed by a pipeline of different processing resources. Important processing resources used in the R2BC include:

1. Tokenizer: The tokenizer subdivides the text superficially into characters or strings categorized as numbers, words (case sensitive) and punctuation. This adds the token annotation to the affected areas, each containing type (number, word, etc.) and length in characters. The processing by the tokenizer should be as efficient as possible and only serve as a basis for future grammatical rules.

2. Gazetteer: A Gazetteer contains relevant proper names from certain areas. It is a nested list containing known proper names for each area (for example, cities, airports, football clubs). Gazetteers are used to recognize domain specific entities, as they are not found in a regular dictionary. The ready-made lists are therefore to be expanded or created individually depending on the document. If a list element is recognized in the text, it gets the lookup annotation by default. However, this can already be changed and determined depending on the list in the Gazetteer.

3. Sentence Splitter: The sentence splitter divides the entire text into individual sentences, which will wrap every sentence in a sentence annotation. This is needed later on for the Part-of-Speech Tagger. The splitter uses a gazetteer with abbreviations to distinguish punctuation, such as question marks, exclamation points, and points from other special characters, such as semicolons and colons.

4. POS-Tagger: The Part-of-Speech tagger annotates all words and symbols according to their grammatical function. No new annotation will be created for this, but the important category feature will be added to the token annotations. This is fundamental for grammatical analysis of sentences. It uses standard dictionaries and rules based on the corpus training of the Wall Street Journal.

Using these standard annotations, very customizable JAPE transducers can be implemented to find and attach the information needed for the specific application. These processing resources operate on the *Java Annotation Patterns Engine* (JAPE), which tests regular expressions of existing annotations and can create further annotations or executes java code according to the result.

JAPE transducers are implemented by JAPE rules consisting of a head (various parameters), a left hand side/LHS (regular expression) and a right hand side/RHS (result upon confirmation of LHS). Therewith, standard annotations the ANNIE system generates can be extended using customized rules in a JAPE transducer. The general composition of the processing pipeline consisting of both text annotation and JAPE transducers is given in [3].

III. Concept

In this chapter we present our concept and tool support for an automated delta analysis. As a first step, we give an overview of the main components of our tool chain consisting of the Requirements to Boilerplates Converter (R2BC) and the Delta Analyzer (DA) as depicted in Fig. 2. According to this concept the R2BC is used to convert random natural language requirements into predefined boilerplates. Once all requirements are available in the predefined syntax, the DA is used to perform an automated delta analysis. In our previous work [1], we present the concept of the R2BC together with the test results of the first implementation. In this chapter, we describe our concept for an automated delta analysis and the architecture of the corresponding tool the DA. Moreover, our concept for an automated delta analysis includes an algorithm for the prioritization of requirements deltas. The description of this algorithm constitutes the third part of this chapter.

A. Methodology of Prototypical Tool Chain

1. Requirements to Boilerplates Converter: The process starts once an OEM submits a CRS to the supplier. In a first step, natural language requirements are translated into predefined boilerplates by the R2BC. After processing, the specification of the OEM is available in a semi-formal language. The semi-formal format offers advantages over natural language. This version of the specifications can then be used for the development process. The aim of the R2BC is therefore to improve the readability or the potential of the requirements through formalization. On this basis, the content of the specifications can then be better understood by the developers at the supplier.

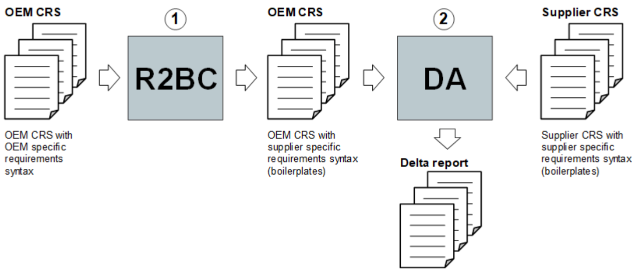

OEM CRS
1
R2BC
OEM CRS
2
DA
Supplier CRS
OEM CRS with OEM specific requirements syntax
OEM CRS with supplier specific requirements syntax (boilerplates)
Delta report
Supplier CRS with supplier specific requirements syntax (boilerplates)

Fig. 2. Methodology for an Automated Delta Analysis

2. Delta Analyzer: These formalized CRSs represent the fundamentals for the approach in this work, as they act as the input for our tool. The custom syntax of requirements allows to compare multiple specifications based on the structure of each requirement. These differences can be identified by the DA and be summarized in a delta report. Differences can then be output for a RE expert on screen or in a file. For this purpose, the program offers procedures for the evaluation of the deltas or metrics for the decision between whether requirements are matching or not. As we describe later, such a categorization can take place based on the annotations generated by the R2BC.

The methodology of the presented tools is shown as a prototypical tool chain in Fig. 2.

B. Delta Analyzer Concept

The Delta Analyzer concept features methods that can identify deltas between old and new CRS, which can then be rated and emphasized for the user. This processing is done for each requirement from the CRS. For an automated finding of suitable requirements in the DA, the requirements must be presented uniformly, which is why we use the boilerplates created by the R2BC (cf. our previous work [1]).

The functions described allow the design of a method for an automated comparison of two specifications in the DA. For this purpose, all requirements from the specification V2 (current, new OEM CRS) are compared with those from V1 (CRS of predecessor product). For each requirement in V2, V1 is traversed to find a corresponding statement there. This leads to the distinction of three categories: (1) identical, (2) comparable with deltas and (3) no comparable was found. In order to help the project members estimating the requirements, giving feedback and offer support to realize major issues to them is an important aspect of our concept. Especially, the impacts of deltas found in (2) has to be rated by our tool. Therefore, metrics were identified and bundled as part of the DA-algorithm to find the best possible match for each requirement. The following procedure was developed:

1. The individual requirements in boilerplates are loaded for specification V1 and V2.
2. For each boilerplate from CRS V2 it is checked whether identical, comparable (with deltas in the specification) or no requirements can be assigned from V1. These three categories are identified based on the GATE annotations from the R2BC. Using further similarity measurement metrics, we can calculate a total degree of correspondence as score.
3. According to the degree of correspondence found, the deltas are identified between the requirements:
   (a) Identical requirement: There was an identical requirement found in both CRS. The request can be accepted without separate consideration.
   (b) Comparable with deltas: Only a partially matching requirement was found. This is likely the case when requirements have changed or been revised. There is a delta between these requirements, that signals a change in the requirements and would therefore mean a change in product specifications. A fine granular breakdown of the distinguishable deltas will be described later.
   (c) New requirement: No matching requirement from V1 could be found. CRS V2 could contain new requirements that were not specified in V1 in any way. This may concern, for example, properties that are defined later in the development process or no information was previously known about.
4. If there are more at least comparable requirements in V1, the best matching requirement is found.
5. The delta report for the user is generated by accessing the results from 4. when V2 was completely traversed.

If deltas have been found, the feasibility of these new requirements must always be checked separately. Different specifications in V2 might decrease the correctness of the requirements. The further correctness of V2 to V1 can not be guaranteed for many deltas, which makes it difficult to realize the requirements. There may be massive differences in specifications, which may result in a partial or complete redesign of the product. As a guideline, the report therefore also contains the overall degree of agreement of the CRS as an absolute and percentage value. The description of the functionality for our tool led to the concrete emergence of three core components. The tool contains components for the mapping of subject logic and GUI as shown in Fig. 3.

The components realize the two main functions 1 and 2 using GATE and Java resources. Based on this concept, the functionalities for each individual component are assigned as follows:

1. NL pre-processing: Since the DA represents the second stage of processing, it uses the results from the R2BC, i.e. the annotations generated there. These will be under access to the program read into the DA. Thus the requirements which are present in boilerplates after R2BC-conversion, are compared by their annotations.

2. Analyzer: The delta analyzer checks the various requirements of specification V2 for their compliance with the requirements V1. The program accesses the loaded annotations of both specifications and compares different requirements. Finally, a delta report is created that clearly displays all found deltas between the requirements.

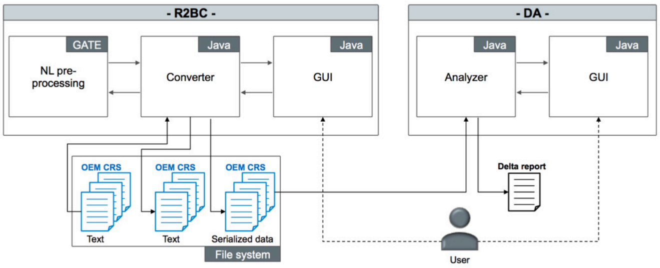


Fig. 3. Components of the Delta-Analyzer

3. GUI: The GUI allows the user to enter the two CRS to be compared. After processing, the user can export the delta report containing the compared requirements.

C. Agreement Level

Aiming to offer robust support in identifying deltas between two requirements, a fine-grained process analyzing each requirement match was designed to give a score to each of these matches. Each single requirement from V2 and V1 is represented as an object in our tool. Every match-object then consists of both found requirements and the agreement level the DA calculated.

The CRS are essentially treated as mathematical sets. For our concept, requirements from V1 and V2 can also be understood as set elements. A *boilerplate-match* then is a triple: $(r2, r1, a) : r2 \in V2, r1 \in V1$, agreement level a. The delta-analyzer aims to rate a match as close to category (a) as possible, searching for an identical requirement in V1 as long as no triple with that agreement level was constructed. The process of scoring the agreement of two requirements consists of two main steps. First, the boilerplate environment is used, analyzing boilerplate-type and then the features (text parts) of each boilerplate. In case of a different type, we assume the requirements as completely different (new requirement in V2, category c), as otherwise they would have shared the same boilerplate-category. If two compared requirements share the same boilerplate-features and category, they are at least comparable (b), if not even identical (a). The task then was to find a method rating several types of deltas all found as (b), as some requirements were partially identical while others were almost distinct.

After this rather rough analysis of R2BC-annotations, we use Levenshtein distance for a deeper, linguistic analysis. The Levenshtein distance calculates the difference between two given strings as the effort (operations) to transform string $s1$ into $s2$ [4]. This method is common when it comes to measuring similarity of two given strings or requirements in our case. After calculating the Levenshtein distance for each component of the requirements matched as (b), we can easily add it to the agreement level contained in the triple together with the requirements. As our tool searches for the best possible match, we can now simply search for the best score comparable requirements have reached.

D. Methodology for the Prioritization of Requirements Deltas

The determination of deltas is an important step to increase efficiency during the RFQ phase. The DA concept includes an additional function to speed-up the work progress during this phase. This function is called requirements delta prioritization. Its main purpose is to rank requirements deltas, which have the highest impact on the development effort, first. As a consequence, project team members can focus on prioritized requirements deltas to determine whether a predecessor component can fulfill the new requirements and if necessary, how much effort is needed to adapt the component. To this end, our requirements delta prioritization algorithm relies on weighting factors, which are defined prior to the delta analysis.

E. Determination of Weighting Factors for Product Features and Functions

Weighting factors indicate how much more development effort is required to adapt a certain feature or function of a product compared to another feature or function. They are a prerequisite for the requirements delta prioritization algorithm and therefore have to be determined first. Our concept for the determination of weighting factors is based on the Analytic Hierarchy Process (AHP) introduced by Thomas L. Saaty [5]. Within this methodology a pair-wise comparison matrix is used to determine which entry is more important than another. In our approach, this determination is executed by the project team within the DA GUI, as depicted in Fig. 4. As a first step, the project team selects a product, for which the weighting factors shall be determined (Fig. 4 Step 1). The selection of a product is important, since weighting factors for certain features and functions can be different for different products. A change in current draw can be of higher impact in a pump, than in an actuator. Second, the project team documents the top ten features and functions of the selected product (Fig. 4 Step 2). We suggest an amount of ten features and functions, but it is also possible to add more. It is sufficient to fill out the headings. The headings for the rows are filled out automatically. According to the application domain, the amount of the main features and functions may vary. The only one boundary condition is that the number of features and functions is finite. During the third step (Fig. 4 Step 3), project team members rate the impact for the development effort on a scale from 1-9, where 1 is equal effort, 2 is low effort, 3 is moderate effort, 4 is moderate plus, 5 is high effort, 6 is high effort plus, 7 is very high effort, 8 is very, very high effort and 9 is extreme effort. We use the scale as suggested by [5] and adjusted the definitions of the numbers, so that they fit our use case. To decrease the effort of filling out the matrix, only the white upper right corner of the matrix must be filled out by the user. The lower left corner is filled out automatically with the reciprocal value by the DA. Once all values have been entered, the DA automatically calculates the geometric mean per feature or function and show them in the respective column in the GUI. The geometric mean figures constitute the weighting factors, which at the same time describe a total order of all features and functions. As a result, the top ten features and functions can be ranked in order of development effort, which is necessary, if one of them shall be adapted in the selected product. Finally, the user can save the weighting factors by pressing "Save weighting factors" (Fig. 4 Step 4).

All weighting factors for a given product are saved in the repository of the DA and can hence be invoked by the requirements delta prioritization algorithm. Since over time the development effort for certain features or functions can change, it is possible to repeat this process. The DA will automatically calculate the mean for all adjustments of weighting factors per feature or function of a product. By this opportunity we aim to receive more accurate weighting factors over time.

F. Prioritization of Requirements Deltas

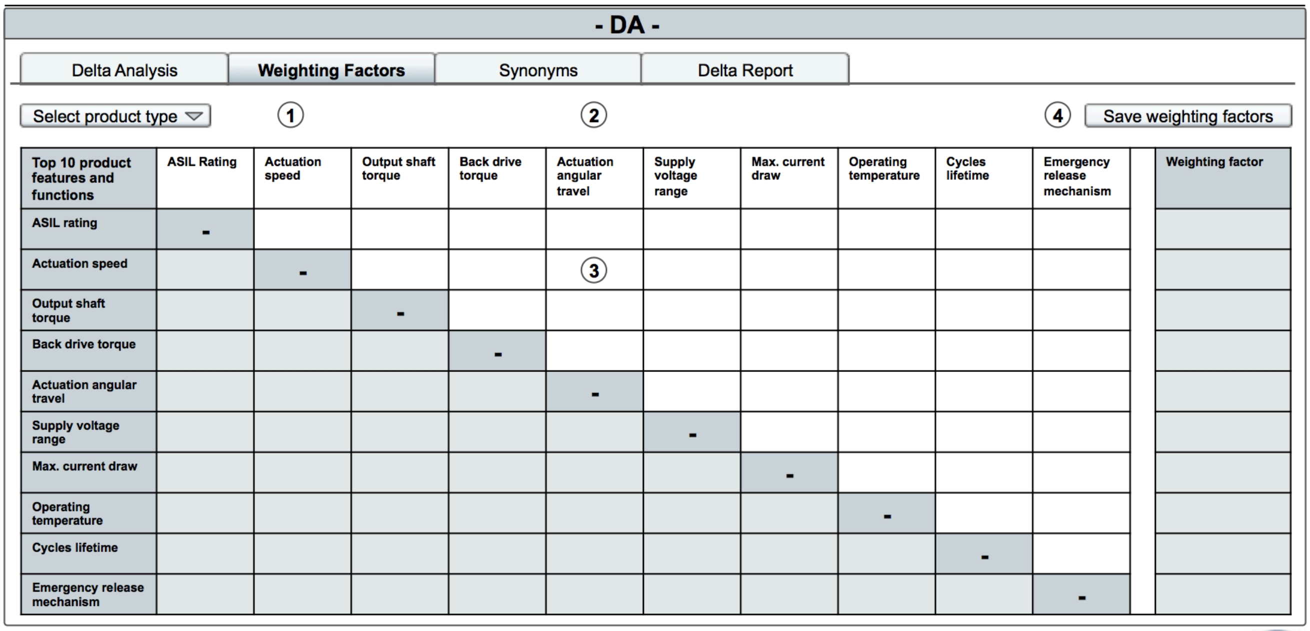
The general aim of the DA is to detect deltas between a supplier CRS and an OEM CRS by comparing requirements sentences with each other. For this purpose, a pairwise comparison of requirements is performed by the DA. Once two requirements address the same system and one of its features or functions, the DA allocates both requirements as a couple in the delta report. As an example, the supplier requirement A and the OEM requirement B constitute the following couple, since both of them address the torque feature of an actuator:

A: The actuator shall provide a stall torque of 50 Nm. B: The waste gate actuator shall provide a torque of 60 Nm.

Since not all requirements from the OEM CRS and the supplier CRS can be allocated as couples, single requirements will exist in the delta report. This is especially true for completely new OEM requirements, which previously were never addressed by the supplier CRS.

Once the delta analysis is finished, the requirements delta prioritization algorithm searches for keywords in all requirements, which are listed in the delta report. The product features and functions, which were documented during the determination of weighting factors, constitute these keywords. If for example the requirements delta prioritization algorithm finds the keyword "torque" in the requirements couple of supplier requirement A and OEM requirement B, this couple will automatically receive the previously defined weighting factor. To this end, the DA invokes the weighting factors from the repository. The DA assigns the same weighting factors to identified requirements couple as well as to single requirements, which could not be allocated to other requirements. To increase the success rate of the weighting factor assignment, the DA provides the user with the possibility to enter synonyms and other words or characters that may be a hint for a certain product feature or function.

Once the prioritization of requirements deltas is finished, the user can rank them by their weight in order to focus on the most heavy-weighted deltas first. However, in case the torque requirement of the OEM does not demand any changes in



Top 10 product features and functions	ASIL Rating	Actuation speed	Output shaft torque	Back drive torque	Actuation angular travel	Supply voltage range	Max. current draw	Operating temperature	Cycles lifetime	Emergency release mechanism		Weighting factor
ASIL rating	-											
Actuation speed		-										
Output shaft torque			-									
Back drive torque				-								
Actuation angular travel					-							
Supply voltage range						-						
Max. current draw							-					
Operating temperature								-				
Cycles lifetime									-			
Emergency release mechanism										-		

Fig. 4. Determination of weighting factors with the pairwise comparison matrix

the current degree of torque, the corresponding requirements couple should have a lower priority for the user, than those couples, which demand changes from the product. To this end, the DA provides a filter function. With this filter, the user can filter for the three categories of requirements couples: identical requirements (no changes of feature or function required), comparable with deltas (changes of feature or function required) new requirements (new feature or function required). With this filter, the user can focus on true requirements deltas with the highest weighting factor.

For those requirements couples, which did not receive a weighting factor, the user may add a factor according to his estimation. Then the prioritization of requirements couples can be repeated. The prioritization order is adapted by the DA automatically. This functionality allows to expand the number of weighting factors for new features and functions and assures that the new weighting factors are related properly to the already present ones. As a consequence, this methodology can be applied to different types of products and can be updated accordingly.

IV. Experiments and Discussion

In this chapter, we present the results of preliminary experiments with the DA prototype. As a first step we describe the target setting and the experimental setup. Second, we present figures from our experiments and discuss the results. Finally, based on the gathered data, we draw conclusions and suggest measures for improvement.

A. Target setting and experimental setup

The target of the experiments is to prove the general feasibility of the DA. The objective of the DA is to determine whether the requirements of the customer CRS are identical, comparable or completely new to the supplier CRS or an previous version of the customer CRS. To assess the performance of the DA, we calculate precision scores for the correct allocation of requirements to the respective categories: identical requirements, comparable with deltas and new requirements.

We conducted four experiments with five real-life CRS. All these specification documents have a common history. CRS 1 is the initial specification. It consists of fragmented sentences and is a rather short document. CRS 2 is a qualitatively improved version of CRS 1. CRS 3, 4 and 5 describe different product generations, which emerged over time. Each CRS contains additional requirements regarding new features and new sub-components. This is a common scenario in the automotive industry. Therefore, we decided to perform the delta analysis with the following couples: CRS 1 with CRS 2, CRS 2 with CRS 3, CRS 1 with CRS 4 and CRS 4 with CRS 5. All CRSs were provided to us as PDF documents.

According to our tool chain, the CRS must be processed by the R2BC first. The reason for this is, that the DA uses information created during the NL pre-processing applied and saved by the R2BC. As a preparatory step we created a new gazetteer list for the R2BC. We used the content of the glossary of terms of all five CRS for our gazetteer and performed the requirements to boilerplates conversion. Also, the JAPE rules for CRS 1, 4 and 5 required some adjustment. The results of the conversion showed a drop of the number of identified requirements for this CRS. We discovered that the requirements authors of these CRS have used “must” instead of “shall” or omitted it completely and applied “to be” or “is” to explain for example an action in their requirements. To this end, we substituted the word “shall” with “must” in our JAPE rules and added “to be” as an alternative explanation. This led

TABLE I
RESULTS OF PRELIMINARY EXPERIMENTS

	Category	Req. per cat.	Correct	Precision	Total req.
CRS 1	Identical requirements	0	0	-	
&	Comparable with deltas	35	33	94.3%	89
CRS 2	New requirements	23	-	-	
CRS 2	Identical requirements	27	25	92.6%	
&	Comparable with deltas	94	93	98.9%	180
CRS 3	New requirements	23	-	-	
CRS 1	Identical requirements	1	1	100%	
&	Comparable with deltas	23	23	100%	61
CRS 4	New requirements	8	-	-	
CRS 4	Identical requirements	3	3	100%	
&	Comparable with deltas	9	9	100%	69
CRS 5	New requirements	25	-	-	

to a high number of requirements conversions.

The gazetteers were mainly used to identify the system name. During preliminary experiments the R2BC converted all requirements sentences into boilerplates, whether the corresponding subjects were system names or other words for example, describing product features. Once the original requirement fit the boilerplate, it was considered useful for the experiments. We also did not correct the results of the R2BC. Some requirements had flaws as a result of the conversion. Since the number of proper requirements was high enough to prove the feasibility of the DA, we considered the incompletely converted requirements negligible.

In our previous work [1], we presented the results of preliminary experiments with the R2BC. In the remainder of this section we focus on the performance of the DA prototype. Within first experiments the DA achieved promising results, as can be seen in Table I.

Within the first experiment with CRS 1 and CRS 2, whereby CRS 2, the DA analyzed 89 requirements from both CRS in less than a second. As a result, the DA identified 33 comparable requirements couples with a precision of 94.3%. Two requirements contained not processable symbols, which lead to error outputs in the delta report. As consequence 33 of the 35 available couples were identified by the DA. All requirements couples, which were listed by the DA in the delta report in the "Comparable with delta" section, mention the same system name respectively. The latter parts of the requirements sentences differ from each other. As can be seen from the numbers, one requirement from CRS 2 can fit several requirements form CRS 1. The DA also found 23 new requirements in CRS 2, which were not listed in CRS 1 before. Yet, the novelty of these requirements is merely proven by their syntactical difference.

As a result of the delta analysis of CRS 2 and CRS 3, the DA identified 25 identical requirements couples with a precision of 92.6%. The precision was reduced by the fact, that two error outputs appeared in the delta report. These were caused by not processable symbols in the original CRS. In the category "Comparable with deltas", the DA found 93 comparable requirements. These include, seven requirements couples, which consisted in their original state of identical sentences per couple. As a matter of fact, these couples should actually be listed in the "Identical requirements" category. But the R2BC added additional words, which stem from the adjacent attribute column in the original CRS to the conversion results of these 14 requirements. Nevertheless, we consider the DA results correct, since the sentences provided by the R2BC differed indeed from each other. As previously mentioned, we defined an agreement level score to determine how equal two requirements are to allocate them as a couple. The observation of experiments with CRS 2 and CRS 3 showed, that requirements with an agreement level score of up to 1.02 differ in only one character. From a score of 1.06 they differ mostly in one word. Once the agreement level score surpasses the mark of 1.5, the requirements are clearly different except for the system name. Finally, the DA found 23 new requirements in CRS 3. Also, here the novelty of the requirements is caused by their syntactic character.

The delta analysis for CRS 1 and 4 resulted in one requirements couple consisting of identical requirements sentences and 23 requirements couples, which are comparable and contain deltas. For both categories the DA achieved 100% precision. For eight requirements from CRS 4, the DA could not identify a comparable requirement in CRS 1 based on the syntactical analysis. In the last experiment with CRS 4 and CRS 5, the DA identified three requirements couples with identical requirements. The identification of identical and comparable requirements worked both with a precision of 100%. In numbers, the DA identified nine requirements couples with comparable requirements and 25 new requirements. In total 399 requirements from five different CRS were analyzed during the experiments.

B. *Discussion and Conclusion*

Preliminary experiments with the DA prototype show sound results. The precision values range from 94.3% to 100%. Within seconds hundreds of requirements can be analyzed and deltas determined. From the number of requirements couples per category can be seen, that some CRS documents are more similar than others. The requirements from CRS 2 and CRS 3 seem to be similar. On the other hand, CRS 1 and CRS 2 seem to be very different.

The major benefit of a delta analysis is to find requirements couples, where both requirements slightly differ from each other. In this case, it is directly clear which feature or function of a component must be adapted. It is also an indication that the supplier already knows how to handle this kind of requirement, since it could be allocated to the supplier requirement or to the requirement of a previous version of the CRS. Especially in the results of the experiment with CRS 2 and CRS 3 this phenomenon can be observed. Requirements couples with an agreement level score of 1.02 differed in only one character. This kind of couples are beneficial, since they show the user of the DA the need for action in a clear matter. For instance, the necessary adjustment of the torque of an actuator could be detected by applying the DA. Scores higher than 1.06 represent requirements couples with requirements, which mostly differ in one word. Beyond the threshold of 1.5 only the system names are equal in both partners of the requirements couple.

Based on these findings we are planning to use the presented thresholds of agreement level scores to highlight requirements couples with slightly different requirements to the user. Furthermore, we plan to evaluate, whether the requirements which were identified as “New requirements” are really new. So far, their categorization was proven only by a syntactic analysis. This evaluation may lead to the understanding, whether a semantic delta analysis would lead to better results. Also, the presented results have shown that the R2BC is still a decisive component of our tool chain and that it has a huge effect on the outcome of the delta analysis. Therefore, we plan to further improve our prototype of the R2BC.

V. Related Work

Since the automated comparison of different data offers a high potential for reduction working hours, a large number of tools for document comparison have been developed. In general, all of these tools are designed to find differences between an original and a corresponding modified document and at the end to describe the identified changes in a third document [6]. However, scientific work on these tools is hard to come by because most of them are proprietary and little to nothing has been published about them. To the best of our knowledge, there is no work that implements an automatic delta analysis similar to our approach.

Schraps and Bosler present an approach to create a requirements ontology by extracting knowledge from software requirements and transferring this knowledge into the ontology. After annotating the requirements using NLP techniques, a pattern recognition algorithm searches for predefined grammatical patterns. Requirements or parts of them fitting the patterns are then integrated into the requirements ontology [7]. While this approach aims at detecting and solving inconsistencies between requirement specifications and software models, the goal of our approach is to find differences and mutualities between multiple requirement specifications.

The Module Comparison Wizard in IBM DOORS can be used to compare two different modules, possibly containing requirement specifications. The tool detects if objects have been added, deleted, edited or if heading numbers have changed. These deltas can be found automatically, they are however of structural or syntactic nature and show no relevance regarding the content similarity or the requirements [8].

The tool compareDocs from DocsCorp aims at comparing two versions of a document and showing differences in form of insertions, deletions and moves. It can be used on different types of documents and event hough it resembles a similar "delta-between-versions" approach as the tool described in this work, the detected deltas have no means of being evaluated regarding their semantic relevance [9].

VI. Summary and Outlook

In this work, we presented our concept for an automated delta analysis, which aims to support project teams in the limited time frame of an RFQ phase. Our proprietary developed prototype the Delta Analyzer (DA) analyses two different requirement specification documents provided in office format and determines the differences between the requirements – namely the deltas. The DA uses information generated by Natural Language Processing techniques applied by our tool R2BC [1]. Preliminary experiments with the DA on real-life requirements specifications yielded good results. The DA allocated requirements into the categories: identical requirements, comparable with deltas and new requirements with a precision of 94.3% to 100%. In addition to that, we introduced our methodology for the prioritization of requirements deltas. This technique is based on the Analytic Hierarchy Process (AHP) [5] and has the purpose of ordering requirements deltas in the delta report according to the development effort for their implementation. This functionality shall enable project teams to focus on the most important need for action first.

The objective of our upcoming research activity is to further improve the performance of the DA. To this end, we will refine the recognition of deltas in the category comparable with deltas. This will allow us to automatically distinguish between more high-level deltas at component level and low-level deltas that relate to specific features or functions of single components. We are currently in the process of implementing the algorithm for prioritizing requirements in our DA prototype. Once this functionality is available, we will test it with different project teams.

References

[1] K. Zichler and S. Helke. R2BC : Tool-Based Requirements Preparation for Delta Analyses by Conversion into Boilerplates. In *Proceedings Workshop on Automotive Software Engineering (ASE 2019)* CEUR-WS, **2308**, pages 45-52, 2019.

[2] R. Weischedel et al. White Paper on Natural Language Processing. In *Proceedings of the Workshop on Speech and Natural Language*. Association for Computational Linguistics, 1989.

[3] D. Thakker and T. Osman and P. Lakin. Gate Jape Grammar Tutorial. Nottingham Trent University, UK, Phil Lakin, UK, Version 1, 2009.

[4] L. Yujian and L. Bo. A Normalized Levenshtein Distance Metric. In *Proceedings IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29.6**, pages 1091-1095, 2007.

[5] Y. Wind and T.L. Saaty. Marketing Applications of the Analytic Hierarchy Process. Management Science **26.7**, pages 641-658, 1980.

[6] D. Massand. Systems and Methods for the Comparison of Annotations within Files U.S. Patent No. 8,732,181. 2014.

[7] M. Schraps and A. Bosler. Knowledge Extraction from German Automotive Software Requirements using NLP-Techniques and a Grammar-based Pattern Detection. In Proc. of the Int. Conf. on Pervasive Patterns and Applications, 2016.

[8] https://www.ibm.com/support/knowledgecenter/en/SSYQBZ_9.6.1/com.ibm.doors.requirements.doc/topics/c_modulecomparisonmarkup.html

[9] https://www.docscorp.com/products/comparedocs/softwaredevelopment-kit-SDK-API

[10] F. Ritter and A. Schul. Entwurf und Implementierung einer Werkzeugunterstützung zur sprachlichen Analyse und automatisierten Transformation von Projektlastenheften im Kontext der Automobilindustrie. Bachelor thesis, FH Dortmund, 2019.

[11] K. Zichler and S. Helke. Ontologiebasierte Abhängigkeitsanalyse im Projektlastenheft. In *Proceedings Automotive - Safety und Security (AUTOMOTIVE 2017)*, GI-LNI, **269**, 2017.

3^{rd} Workshop on Internet of Things—Enablers, Challenges and Applications

THE Internet of Things is a technology which is rapidly emerging the world. IoT applications include: smart city initiatives, wearable devices aimed to real-time health monitoring, smart homes and buildings, smart vehicles, environment monitoring, intelligent border protection, logistics support. The Internet of Things is a paradigm that assumes a pervasive presence in the environment of many smart things, including sensors, actuators, embedded systems and other similar devices. Widespread connectivity, getting cheaper smart devices and a great demand for data, testify to that the IoT will continue to grow by leaps and bounds. The business models of various industries are being redesigned on basis of the IoT paradigm. But the successful deployment of the IoT is conditioned by the progress in solving many problems. These issues are as the following:

- The integration of heterogeneous sensors and systems with different technologies taking account environmental constraints, and data confidentiality levels;
- Big challenges on information management for the applications of IoT in different fields (trustworthiness, provenance, privacy);
- Security challenges related to co-existence and interconnection of many IoT networks;
- Challenges related to reliability and dependability, especially when the IoT becomes the mission critical component;
- Zero-configuration or other convenient approaches to simplify the deployment and configuration of IoT and self-healing of IoT networks;
- Knowledge discovery, especially semantic and syntactical discovering of the information from data provided by IoT;

The IoT conference is seeking original, high quality research papers related to such topics. The conference will also solicit papers about current implementation efforts, research results, as well as position statements from industry and academia regarding applications of IoT. The focus areas will be, but not limited to, the challenges on networking and information management, security and ensuring privacy, logistics, situation awareness, and medical care.

TOPICS

The IoT conference is seeking original, high quality research papers related to following topics:

- Future communication technologies (Future Internet; Wireless Sensor Networks; Web-services, 5G, 4G, LTE, LTE-Advanced; WLAN, WPAN; Small cell Networks…) for IoT,
- Intelligent Internet Communication,
- IoT Standards,
- Networking Technologies for IoT,
- Protocols and Algorithms for IoT,
- Self-Organization and Self-Healing of IoT Networks,
- Trust, Identity Management and Object Recognition,
- Object Naming, Security and Privacy in the IoT Environment,
- Security Issues of IoT,
- Integration of Heterogeneous Networks, Sensors and Systems,
- Context Modeling, Reasoning and Context-aware Computing,
- Fault-Tolerant Networking for Content Dissemination,
- Architecture Design, Interoperability and Technologies,
- Data or Power Management for IoT,
- Fog—Cloud Interactions and Enabling Protocols,
- Reliability and Dependability of mission critical IoT,
- Unmanned-Aerial-Vehicles (UAV) Platforms, Swarms and Networking,
- Data Analytics for IoT,
- Artificial Intelligence and IoT,
- Applications of IoT (Healthcare, Military, Logistics, Supply Chains, Agriculture, …),
- E-commerce and IoT.

The conference will also solicit papers about current implementation efforts, research results, as well as position statements from industry and academia regarding applications of IoT. Focus areas will be, but not limited to above mentioned topics.

EVENT CHAIRS

- **Cao, Ning,** College of Information Engineering, Qingdao Binhai University
- **Furtak, Janusz,** Military University of Technology, Poland
- **Hodoň, Michal,** University of Žilina, Slovakia
- **Zieliński, Zbigniew,** Military University of Technology, Poland

PROGRAM COMMITTEE

- **Al-Anbuky, Adnan,** Auckland University of Technology, New Zealand
- **Antkiewicz, Ryszard,** Military University of Technology, Poland
- **Baranov, Alexander,** Russian State University of Aviation Technology, Russia

- **Brida, Peter,** University of Zilina, Slovakia
- **Chudzikiewicz, Jan,** Military University of Technology in Warsaw, Poland
- **Cui, Huanqing,** Shandong University of Science and Technology, China
- **Dadarlat, Vasile-Teodor,** Univiversita Tehnica Cluj-Napoca, Romania
- **Ding, Jianrui,** Harbin Institute of Technology, China
- **Diviš, Zdenek,** VŠB-TU Ostrava, Czech Republic
- **Fortino, Giancarlo,** Università della Calabria
- **Fouchal, Hacene,** University of Reims Champagne-Ardenne, France
- **Fuchs, Christoph,** Fraunhofer Institute for Communication, Information Processing and Ergonomics FKIE, Germany
- **Giusti, Alessandro,** CyRIC - Cyprus Research and Innovation Center, Cyprus
- **Hudik, Martin,** University of Zilina
- **Husár, Peter,** Technische Universität Ilmenau, Germany
- **Johnsen, Frank T.,** Norwegian Defence Research Establishment (FFI), Norway
- **Jurecka, Matus,** University of Žilina, Slovakia
- **Kafetzoglou, Stella,** National Technical University of Athens, Greece
- **Kapitulík, Ján,** University of Žilina, Slovakia
- **Karastoyanov, Dimitar,** Bulgarian Academy of Sciences, Bulgaria
- **Karpiš, Ondrej,** University of Žilina, Slovakia
- **Kochláň, Michal,** University of Žilina, Slovakia
- **Krco, Srdjan,** DunavNET
- **Laqua, Daniel,** Technische Universität Ilmenau, Germany
- **Lenk, Peter,** NATO Communications and Information Agency, Other
- **Li, Guofu,** University of Shanghai for Science and Technology, China
- **Marks, Michał,** NASK - Research and Academic Computer Network, Poland
- **Milanová, Jana,** University of Žilina, Slovakia
- **Monov, Vladimir V.,** Bulgarian Academy of Sciences, Bulgaria
- **Murawski, Krzysztof,** Military University of Technology, Poland
- **Niewiadomska-Szynkiewicz, Ewa,** Research and Academic Computer Network (NASK), Institute of Control and Computation Engineering, Warsaw University of Technology
- **Ohashi, Masayoshi,** Advanced Telecomunications Research Institute International / Fukuoka University, Japan
- **Papaj, Jan,** Technical university of Košice, Slovakia
- **Ramadan, Rabie,** Cairo University, Egypt
- **Ševčík, Peter,** University of Žilina, Slovakia
- **Shaaban, Eman,** Ain-Shams university, Egypt
- **Shu, Lei,** Guangdong University of Petrochemical Technology, China
- **Skarmeta, Antonio,** University of Murcia
- **Smirnov, Alexander,** Linux-WSN, Linux Based Wireless Sensor Networks, Russia
- **Staub, Thomas,** Data Fusion Research Center (DFRC) AG, Switzerland
- **Suri, Niranjan,** Institute of Human and Machine Cognition
- **Teslyuk, Vasyl,** Lviv Polytechnic National University, Ukraine
- **Wang, Zhonglei,** Karlsruhe Intitute of Technology, Germany
- **Wrona, Konrad,** NATO Communications and Information Agency
- **Xiao, Yang,** The University of Alabama, United States
- **Zhang, Tengfei,** Nanjing University of Post and Telecommunication, China

DOI: 10.15439/2019F348
ISSN 2300-5963 ACSIS, Vol. 19

A Fog Computing Architecture for Security and Quality of Service

Bruno Nunes Barreto
Federal University of Sergipe
Av. Marechal Rondon, s/n,
Sao Cristovao, Sergipe, Brazil
Email: bnbarreto@gmail.com

Alexandre Rezende de Sa
Federal University of Sergipe
Av. Marechal Rondon, s/n,
Sao Cristovao, Sergipe, Brazil
Email: alexandresa@ufs.br

Admilson de Ribamar Lima Ribeiro
Federal University of Sergipe
Av. Marechal Rondon, s/n,
Sao Cristovao, Sergipe, Brazil
Email: admilson@ufs.br

***Abstract*—The Fog Computing paradigm is an emerging architecture and focuses on optimizing resources for the Internet of Things environment, bringing to the Edge, Cloud's characteristics. The demand generated by the number of possible devices in this network attracts problems related to quality of service, security, among others, attracting researchers from the most diverse areas. In our work, in addition to performing a study on selected works in a mapping process, detecting trends in the use of Fog architectures. The main contribution is presented by a security-based Fog Computing architecture using QoS for scalable environments with Docker containers for orchestration and deployment of security with SDN.**

I. Introduction

THE TECHNOLOGICAL evolution of embedded equipment has enabled virtual communication with certain objects so that we can manage and operate them at a distance through the Internet. With a finality of increase the interactional capacity in systems, a new paradigm called generically Internet of Things (IoT) has been emerging [1].

Through the integration of the most varied technologies, it aims to enable network communication between people, objects and things with different levels of autonomy, extracting and / or providing services and information among themselves or to other devices through the Internet. The IoT architecture can be treated as a physical, virtual or hybrid system, being able to make use of technologies such as Cloud Computing [2], able to overcome the limitations of computing and storage in intelligent devices, besides providing elastic resources to them [3]. According [3], [4] and [5], due to the need to support mobility, geographical distribution, location recognition and low latency demand for some applications, the Cloud meet with some difficulties.

To overcome these difficulties, Cloud features were brought to the edge of the network [6], [7] and [8], thus forming Fog Computing, or simply Fog, which, as a link between IoT and Cloud, induces the extra functionalities required for specific processing of applications, such as filtering and aggregation, before transferring the data to the Cloud [9].

Taking advantage of IoT's capabilities, a wide range of intelligent solutions and applications for the most diverse areas, such as Smart City and Smart Home, have been proposed and increasingly demanded of it, with forecast growth in equipment usage to 50 billion units by 2022, including sensors and actuators [10]. However, this generation advance has been presented in a highly complex way, which has been demanding and moving researchers from the most varied areas of knowledge, besides the need to create environments for the performance analysis of these studies. Some challenges of Fog are listed by [5] and [11], which consider the importance of identifying appropriate techniques and metrics for efficient resource provisioning and management.

In [12], they states that a large number of links and different interactions between edge nodes in IoT makes it a complex and scalable system; therefore, it is difficult to achieve the dynamic requirements of Quality of Service (QoS). How described in [13] and [11] argue that the absence of Service Level Agreement (SLA) management, as well as sustainable metrics, make it difficult to maintain a QoS acceptable in highly dynamic environments. This increase in the number of devices on the network also creates security-related issues, making these endpoints an easy target for malicious people to compromise these devices for use in large-scale attacks.

Thus, our paper aims to present a Fog Computing architecture to provide QoS and security through an orchestrated and virtualized environment, including characteristics such as interoperability and scalability.

The remainder of this paper is structured as follows: Section 2 describes the Fog Computing architeture applied in this study. Next, section 3 presents the implementation issues, followed by related works in the section 4. Section 5 presents a Conclusion.

II. Fog Computing Architecture

According to the paper presented by [14], six criteria considered important for the Fog Computing architecture are: Heterogeneity, QoS Management, Scalability, Mobility, Federation and Interoperability. The architecture we propose next, not only to meet some of these criteria, as well as aspects related to security, providing a consistent, manageable, and secure environment with characteristics that may facilitate the commercialization of services implemented.

This study was financed in part by the Coordenacao de Aperfeicoamento de Pessoal de Nivel Superior - Brasil (CAPES) - Finance Code 001.

These approaches, when contemplated by other articles, are solved individually or in smaller numbers, as we can observe in topic IV (related works). In addition, we are not aware of the use of the K-means algorithm in a Fog Computing.

The proposal of our paper is based on the use of a three-layer architecture similar to that proposed by [4], concomitantly contemplating the six functional blocks of IoT presented by [2] (devices, comunication, services, management, security and application).

As we can see from the Fig. 1, the architecture presents the layers in a well-defined way, where we have the tradicional Cloud at one end, the Fog at the interim layer (composed by Fog nodes) and the edge with the IoT devices. The IoT devices are one of the aforementioned functional blocks, being sensors, actuators, smartphones, among others capable of generating and consuming Fog data.

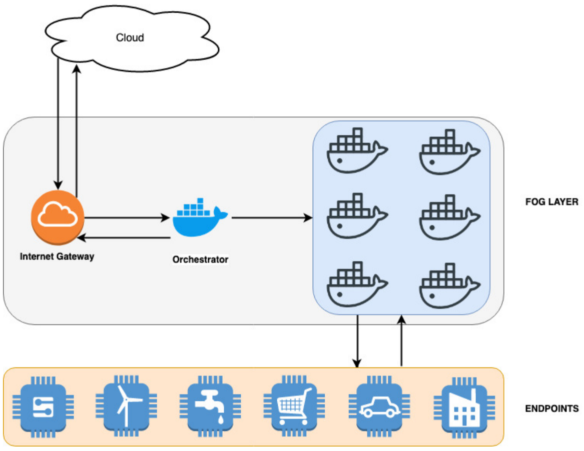


Fig. 1. Fog Archteture proposal.

Interoperability between layers and devices is achieved through the functional block of communication, by means of the data links and their virtualized infrastructure, since much of it is based on Docker containers. Although there are other solutions that promote the use of containers, the Docker offers fault tolerance, service management and deployment capabilities that facilitate the solution delivery process.

Virtualization is a strong trend in the implementation of Fog Computing architectures as we will see in the related works of this paper, making it possible to meet the federation criterion if it becomes a standard used by other service providers, in addition, it makes the architecture scalable through use of a swarm structure, allowing to act in order to deliver the solution continuously orchestrated, attending to the service block.

The last three functional blocks (management, security, and application) are served by another strong feature of this architecture, which is being presented at a time when the threat detection models begin to act directly in Fog layer, allowing the time to decision making is reduced as internal and external threats are identified, thus improving QoS.

This structure will rely on the use of an unsupervised artificial intelligence algorithm capable of learning about anomalies and behavior (DDoS) in a distributed way, which is one of the ten major security flaws in a Fog architecture, according to [15].

As can be seen in the work of [16] and [17], the use of the K-means algorithm presented a very high hit rate compared to other techniques. This algorithm will run in the Cloud (Fig. 2) for training and validation of the samples. Will learn by behavior patterns from open source datasets and then send information to the orchestrator at Fog Computing who will be responsible for generating metrics about the environment as well as resource provisioning computational linked to the models learned in the Cloud.

The orchestrator registers the status of all fog nodes, including the activation and disconnection of nodes, the type of nodes and the IP address of each of them, and is responsible for managing the resources of those nodes that will communicate with the endpoints.

The resources management, among its characteristics, will enable the architecture to simultaneously meet the demands of applications that have or do not have restrictions in real time, prioritizing the guarantee of resources for the most needed or that there is an SLA contract with the client.

The model is combined with the use of SDN (Software Defined Networking) devices since it will be responsible for performing the traffic routing to the endpoints, as well as assisting in the detection, since these data are processed by the Fog node and the cluster, thus providing a better distribution of responsibilities and lower latency among taxpayers. The gateway aims to effect separation and translation between the external and internal networks.

III. Implementation Issues

In order for the environment to achieve the objectives proposed by our architecture, we have a hardware and software structure that will be described as follows:

For anomaly processing solution will be used an Amazon Web Services (AWS) as Cloud Computing Services to find patterns of DDoS attacks. The displayed gateway will be set by a raspberry pi 3 device running the Raspbian Stretch Lite operating system. In order to be orchestrated, 2 physical machines configured with 3.2 Ghz i5 processors and 8 GB ram DDR3 memory will be used, running the linux operating system in the debian 9.9 distribution, as well as the Docker Community edition in version 17.12.1-ce where the portainer management configured, with the portainer/portainer image being available in the Docker hub, chosen for this experiment.

This tool contributes the orchestration of the services in a facilitated way through the use of the webhooks, increasing the practicality in the process of automation of deploy of the final application in the containers that will be destined to solution of SDN.

The area marked in blue in Fig. 2 is responsible for manipulating Fog traffic with IoT devices, applying the rules learned through the cloud and identifying it as malicious or not according to its characteristics. Considering that the entire decision-making process should be automated, this environment is supported by SDN, responsible for providing the necessary intelligence and automation, creating intelligent routes according to the context.

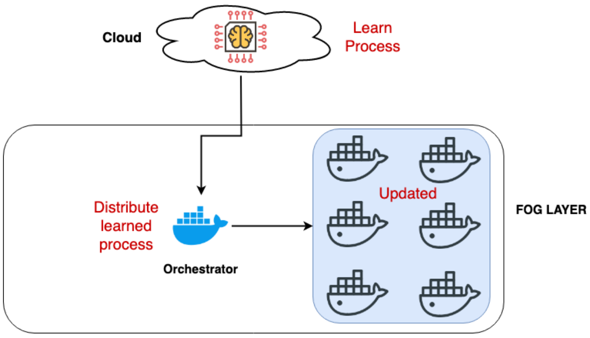


Fig. 2. Fog Security Service.

IV. Related Work

In this session, we discussed the studies considered relevant to our work and commented on the tools, devices, and algorithms most used by them, aiming at a better view of trends and how research in the field of Fog Computing has been developing.

Most of the papers were identified through the systematic mapping process proposed by [18], where we considered the Fog Computing architectures approach that involved both quality of service issues including performance analysis for intelligent environments, as well as questions of security. In this way, we analyze these issues separately in order to facilitate understanding.

A. Related works to architecture in QoS context:

In [19], is present a layered architecture called Fog-to-Cloud (F2C) and compare with an optimized F2C (OF2C) and the traditional Cloud, presenting through simulation and use case in health care the benefits of running services in the different F2C layers. As a result, using the Tareador and Paraver tools, the authors demonstrate an improvement in the task execution speed of 32,05% of the OF2C architecture in relation to the traditional Cloud and poses as a challenge the creation of resource management strategies in different layers of F2C to provide QoS.

A implement through simulation with a framework called Stack4Things, structure based on OpenStack that includes IaaS (Infrastructure as a Service) and PaaS (Platform as a Service) is presented by [20]. They also present a case study of environmental data collection through #SmartME (project to stimulate the creation of a new virtual ecosystem of smart city for the Messina's city). This work indicates anothers types of services that will be provided by Fog, reinforcing the suggestion made by [11] that adapting the Cloud SLA to Fog Computing may be a possible solution for the implementation of this agreement and also as an aid to QoS.

The efficient sharing of client network resources covered by [21], creates network layers configured using SDN and VNF deployed on low-cost common network devices (EX: Raspberry) to approximate wireless and custom services of mobile devices and sensors . As a result, the average cloud delay was approximately 133 ms, versus 12 and 5.3 ms for single board and PC computers, respectively. The environment configured in this work is approximated with the outline of our proposal.

An anomaly detection solution for the smart city application based on Fog, connected to LPWAN and evaluated through algorithms in the testbed of the city of Antwerp is proposed by [22]. The results show that both the Birch cluster and the RC anomaly detection mechanisms can be executed by Fog features. The LPWAN technologies evaluated and validated for the application of air quality were: IEEE 802.11ah, DASH7 and LTE-M.

In [23] study the issue of resource continuity and coordinated Fog and Cloud management and propose the fundamental blocks for system architecture. They demonstrate the benefits of a layer management approach by considering the size and time to search for smart city databases. The authors observed that the smaller the city area the smaller the database size, the lookup time, the lower the number of services to be executed, and thus the lower the interest in these services by the users.

The use of the SDN architecture in a Fog Computing architecture is proposed by [24], focusing on real-time vehicle traffic management, seeking performance enhancement and improved traffic management and QoS in real-time data distribution. In this work, an architecture similar to the one proposed in this paper is used, but its objective is to use it in a vehicular environment.

B. Works related to security issues:

Presented by [25] on his work about Deep Learning on despite the success of traditional Internet cryptographic solutions, factors such as system development flaws, increased attack surfaces, and hacking skills have proven the inevitability of detection mechanisms. Traditional approaches to machine-based attack detection have been successful in the last few decades, but it has already been proven that they have low accuracy and less scalability for detecting cyber attacks on massively distributed nodes such as IoT. The proliferation of deep learning and technological advancement of hardware can pave the way for the detection of the current level of sophistication of cyber attacks in high-end networks. The application of deep networks has already been successful in large areas of data, and this indicates that end-to-end computing may be the ultimate beneficiary of the attack detection approach because a large amount of data produced by IoT devices that deep models learn better than surface algorithms and showed that Deep Learning (DL) models perform well when using unsupervised learning in Zero Day applications, improving model accuracy in invisible and mutant attacks.

In [26] has defined Fog Computing as a new paradigm with many different features of Cloud Computing. Because features are limited, Mobile Edge Computing (MEC) Fog nodes / hosts are vulnerable to cyber attacks. IDS is a fundamental technique for solving the problem. As the Extreme Learning Machine (ELM) has the characteristics of rapid training speed and good generalization capability, a new light IDS called the extreme selection machine (SS-ELM) is presented. The reason why this new model is proposed is justified because the Fog nodes / MEC hosts do not have the capacity to store extremely large amounts of training data sets. Thus, they are stored, calculated and sampled by the Cloud servers. Then the selected sample is supplied to the Fog / MEC hosts for training. This design can reduce training time and increase detection accuracy. The experimental simulation verifies if the SS-ELM shows good intrusion detection performance in terms of accuracy, training time and receiver operating characteristic (ROC) value.

According to the work of [27] as proposed for IoT applications in which it uses Fog Computing to implement an intrusion detection based on the distributed model. The proposed system consists of two modules: Detection of Fog node attacks and summarization on a Cloud server. In this work the Extreme Learning Machine (ELM) algorithm was used and from it a variant called Online Sequential ELM (OS-ELM) was created to identify the attacks in the inbound traffic of IoT virtual clusters.

In [28] proposed to use the deep learning approach to understand that for the treatment of a large data demand, this algorithm is resilient against metamorphosis attacks with high detection accuracy. In this work it is proposed the use of an LSTM network for detecting distributed cyber attacks in Fog communication for things. The experiments conducted demonstrate the effectiveness and efficiency of deeper models compared to traditional models of machine learning.

As shown in the article of [16], it was proposed a DDoS detection model with K-Means algorithm customization that compared to other works provided a higher rate of detection of anomalies, taking into account factors such as True Positive Rate, False Positive Rate and Recall Rate. In addition, is used the main Open Source Dataset (DARPA, CAIDA, CICIDS), as well as the real-world dataset to proposed benchmark. It forms very high hit rates compared to related jobs.

V. Conclusion

The systematic mapping process used in this paper was extremely important for the direction in the search for the state of the art, resulting in the theoretical basis and the identification of the current conjuncture of Fog computing as a whole reported in this paper through the introduction and related works. This corroborated for a better view of the architectural tendencies, devices and tools used in a Fog environment, such as we also indicate in our work. The virtualization and testbeds, for example, are quite common in the environment in question.

In this paper, we present a Fog Computing architecture capable of providing a consistent, manageable, secure environment with specific characteristics relevant to a Fog and to IoT, such as interoperability, scalability, management, among others. This is due to the fact that we have used a virtualized, orchestrated and intelligent environment, a structure that can facilitate the service delivery process between the existing layers in a secure way. The ability to replicate internal security in an agile way is another important aspect to note.

As future work, this architecture model can be validated through simulations, emulations or even applied in production environments, since in the presented model the SDN was used in the application mode through the software Open vSwitch, however, it is interesting to substitute this model by a professional SDN switch. Taking into consideration that the object of study of our work is Fog and its operation, it was not taken into account the fact of security problems in Cloud Computing, and this issue should be treated in another paper with this focus.

The QoS covered in our work makes it possible the service guarantee, contributing to the business aspect of Fog Computing, which is usually the service level agreement (SLA), negotiated between the service provider and the client, but the commercialization of services involving the Fog still need to be researched.

References

[1] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010. doi: 10.1016/j.comnet.2010.05.010. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2010.05.010

[2] P. P. Ray, "A survey on Internet of Things architectures," *Journal of King Saud University - Computer and Information Sciences*, vol. 30, no. 3, pp. 291–319, 2016. doi: 10.1016/j.jksuci.2016.10.003. [Online]. Available: https://doi.org/10.1016/j.jksuci.2016.10.003

[3] S. Yi, C. Li, and Q. Li, "A Survey of Fog Computing," *Proceedings of the 2015 Workshop on Mobile Big Data - Mobidata '15*, no. June 2015, pp. 37–42, 2015. doi: 10.1145/2757384.2757397. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2757384.2757397

[4] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and Its Role in the Internet of Things," pp. 13–15, 2012.

[5] R. Mahmud, R. Kotagiri, and R. Buyya, "Fog Computing: A Taxonomy, Survey and Future Directions," pp. 1–28, 2016. doi: 10.1007/978-981-10-5861-5_5. [Online]. Available: http://arxiv.org/abs/1611.05539%0Ahttp://dx.doi.org/10.1007/978-981-10-5861-5_5

[6] P. K. Sharma, M. Y. Chen, and J. H. Park, "A Software Defined Fog Node Based Distributed Blockchain Cloud Architecture for IoT," *IEEE Access*, vol. 6, pp. 115–124, 2018. doi: 10.1109/ACCESS.2017.2757955

[7] Y. Ai, M. Peng, and K. Zhang, "Edge computing technologies for Internet of Things: a primer," *Digital Communications and Networks*, vol. 4, no. 2, pp. 77–86, 2018. doi: 10.1016/j.dcan.2017.07.001. [Online]. Available: https://doi.org/10.1016/j.dcan.2017.07.001

[8] K. Vohra and M. Dave, "Multi-Authority Attribute Based Data Access Control in Fog Computing," *Procedia Computer Science*, vol. 132, pp. 1449–1457, 2018. doi: 10.1016/j.procs.2018.05.078. [Online]. Available: https://doi.org/10.1016/j.procs.2018.05.078

[9] F. Al-Doghman, Z. Chaczko, A. R. Ajayan, and R. Klempous, "A review on Fog Computing technology," *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 001 525–001 530, 2016. doi: 10.1109/SMC.2016.7844455. [Online]. Available: http://ieeexplore.ieee.org/document/7844455/

[10] K. Yasumoto, H. Yamaguchi, and H. Shigeno, "Survey of Real-time Processing Technologies of IoT Data Streams," *Journal of Information Processing*, vol. 24, no. 2, pp. 195–202, 2016. doi: 10.2197/ipsjjip.24.195

[11] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 1, pp. 416–464, 2018. doi: 10.1109/COMST.2017.2771153

[12] L. Li, S. Li, and S. Zhao, "QoS-Aware scheduling of services-oriented internet of things," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1497–1507, 2014. doi: 10.1109/TII.2014.2306782

[13] R. Prodan and S. Ostermann, "A survey and taxonomy of infrastructure as a service and web hosting cloud providers," *Proceedings - IEEE/ACM International Workshop on Grid Computing*, pp. 17–25, 2009. doi: 10.1109/GRID.2009.5353074

[14] C. Mouradian, D. Naboulsi, S. Yangui, R. H. Glitho, M. J. Morrow, and P. A. Polakos, "A Comprehensive Survey on Fog Computing: State-of-the-Art and Research Challenges," *IEEE Communications Surveys and Tutorials*, vol. 20, no. 1, pp. 416–464, 2018. doi: 10.1109/COMST.2017.2771153

[15] A. Aljumah and T. A. Ahanger, "Fog computing and security issues: A review," in *2018 7th International Conference on Computers Communications and Control (ICCCC)*, May 2018. doi: 10.1109/ICCCC.2018.8390464 pp. 237–239.

[16] Y. Gu, K. Li, Z. Guo, and Y. Wang, "Semi-supervised k-means ddos detection method using hybrid feature selection algorithm," *IEEE Access*, vol. PP, pp. 1–1, 05 2019. doi: 10.1109/ACCESS.2019.2917532

[17] M. I. W. Pramana, Y. Purwanto, and F. Y. Suratman, "Ddos detection using modified k-means clustering with chain initialization over landmark window," in *2015 International Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, Aug 2015. doi: 10.1109/ICCEREC.2015.7337056 pp. 7–11.

[18] K. Petersen, R. Feldt, S. Mujtaba, and M. Mattsson, "Systematic Mapping Studies in Software Engineering," *12Th International Conference on Evaluation and Assessment in Software Engineering*, vol. 17, p. 10, 2008. doi: 10.1142/S0218194007003112. [Online]. Available: http://www.cse.chalmers.se/~feldt/publications/petersen_ease08_sysmap_studies_in_se.pdf

[19] X. Masip-Bruin, E. Marï£¡n-Tordera, G. Tashakor, A. Jukan, and G. Ren, "Foggy clouds and cloudy fogs: a real need for coordinated management of fog-to-cloud computing systems," *IEEE Wireless Communications*, vol. 23, no. 5, pp. 120–128, 2016. doi: 10.1109/MWC.2016.7721750

[20] D. Bruneo, S. Distefano, F. Longo, and G. Merlino, "An IoT Testbed for the Software Defined City Vision: The #SmartMe Project," in *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, 2016. doi: 10.1109/SMARTCOMP.2016.7501678 pp. 1–6.

[21] M. S. Carmo, S. Jardim, A. V. Neto, R. Aguiar, and D. Corujo, "Towards fog-based slice-defined WLAN infrastructures to cope with future 5G use cases," in *2017 IEEE 16th International Symposium on Network Computing and Applications (NCA)*, 2017. doi: 10.1109/NCA.2017.8171397 pp. 1–5.

[22] J. Santos, P. Leroux, T. Wauters, B. Volckaert, and F. D. Turck, "Anomaly detection for Smart City applications over 5G low power wide area networks," in *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, 2018. doi: 10.1109/NOMS.2018.8406257. ISSN 2374-9709 pp. 1–9.

[23] X. Masip-Bruin, E. Marin-Tordera, A. Jukan, and G.-J. Ren, "Managing resources continuity from the edge to the cloud: Architecture and performance," *Future Generation Computer Systems*, vol. 79, pp. 777–785, 2018. doi: https://doi.org/10.1016/j.future.2017.09.036. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167739X17302686

[24] K. S. Sahoo and B. Sahoo, "SDN Architecture on Fog Devices for Realtime Traffic Management : A Case Study SDN architecture on fog devices for realtime traffic management : A case study," no. October, 2017. doi: 10.1007/978-81-322-3592-7

[25] A. Abeshu and N. Chilamkurti, "Deep Learning: The Frontier for Distributed Attack Detection in Fog-To-Things Computing," *IEEE Communications Magazine*, vol. 56, no. 2, pp. 169–175, 2018. doi: 10.1109/MCOM.2018.1700332

[26] X. An, X. Zhou, X. Lü, F. Lin, and L. Yang, "Sample selected extreme learning machine based intrusion detection in fog computing and MEC," *Wireless Communications and Mobile Computing*, vol. 2018, pp. 1–10, 2018. doi: 10.1155/2018/7472095

[27] S. Prabavathy, K. Sundarakantham, and S. M. Shalinie, "Design of cognitive fog computing for intrusion detection in Internet of Things," *Journal of Communications and Networks*, vol. 20, no. 3, pp. 291–298, 2018. doi: 10.1109/JCN.2018.000041

[28] A. Diro and N. Chilamkurti, "Leveraging LSTM Networks for Attack Detection in Fog-to-Things Communications," *IEEE Communications Magazine*, vol. 56, no. 9, pp. 124–130, 2018. doi: 10.1109/MCOM.2018.1701270

Advances in Information Systems and Technology

IST is a FedCSIS conference track aiming at integrating and creating synergy between FedCSIS technical sessions that thematically subscribe to the disciplines of information technology and information systems. The track emphasizes the issues relevant to information technology and necessary for practical, everyday needs of business, other organizations and society at large. This track takes a sociotechnical view on information systems and relates also to ethical, social and political issues raised by information systems. Technical sessions that constitute IST are:

- AITM'19—16th Conference on Advanced Information Technologies for Management
- DSH'19—1st Special Session on Data Science in Health
- InC2Eco'19—1st Workshop on Data Analysis and Computation for Digital Ecosystems
- ISM'19—14th Conference on Information Systems Management
- KAM'19—25th Conference on Knowledge Acquisition and Management

1^{st} Special Session on Data Science in Health

THE Special Session on Data Science in Health is a forum on all forms of data analysis, health economics, information systems and data based health service research, focusing mainly on the interaction of those four fields. Here, data-driven solutions can be generated by understanding complex real-world health related problems, critical thinking and analytics to derive knowledge from (big) data. The past years have shown a forthcoming interest on innovative data technology. Already now we can see how immense amounts of data and rapidly increasing, inexpensive computing power will lead the world to base its decisions more and more on data. We therefore have to work together. We need the knowledge of researchers from different fields applying diverse perspectives and using different methodological directions to find a way to grasp and fully understand the power and opportunities of big data in health.

This special session is a joint track by WIG2, the Scientific Institute for health economics and health service research, and the Information Systems Institute of Leipzig University.

TOPICS

We embrace a rich array of issues on data science in health and offer a platform for research from diverse methodological directions, including quantitative empirical research as well as qualitative contributions. We welcome research from a medical, technological, economic, political and societal perspective.

The topics of interest therefore include but are not limited to:

- Data analysis in health
- Health Data management
- Health economics
- Data based health service research
- Integrating data in integrated care
- AI in integrated care
- Spatial health economics
- Structural equation modelling in medical research
- Risk adjustment and Predictive modelling

EVENT CHAIRS

- **Franczyk, Bogdan,** University of Leipzig, Germany
- **Häckl, Dennis,** WIG2 Institute for health economics and health service research, Leipzig, Germany

PROGRAM COMMITTEE

- **Alpkoçak, Adil,** Dokuz Eylul University
- **Dey, Nilanjan,** Techno India College of Technology, India
- **Kossack, Nils,** Head Mathematics and Statistics, WIG2 Institute for Health Economics and Health Service Research
- **Kozak, Karol,** Fraunhofer and Uniklinikum Dresden, Germany
- **Militzer-Horstmann, Carsta,** WIG2 Institute for Health Economics and Health Service Research, Information Systems Institute of the University of Leipzig, Germany
- **Popowski, Piotr,** Medical University of Gdańsk, Poland
- **Sachdeva, Shelly,** National Institute of Technology Delhi
- **Wasielewska-Michniewska, Katarzyna,** Systems Research Institute of the Polish Academy of Sciences, Poland
- **Wende, Danny,** WIG2 Institute for Health Economics and Health Service Research And Technical University Dresden

Position Papers of the Federated Conference on Computer Science and Information Systems pp. 79–82 DOI: 10.15439/2019F354 ISSN 2300-5963 ACSIS, Vol. 19

Maximum Simulated Likelihood: Don't Stop 'Til You Get Enough?

Christopher Schrey*†, Tobias Schäffer*‡, Carsta Militzer-Horstmann*† and Nils Kossack*
*WIG2 Institute, Leipzig
†Leipzig University
‡Martin Luther University of Halle-Wittenberg

***Abstract*—Maximum simulated likelihood estimation can be employed in empirical health economics, amongst others, to tackle issues concerning endogenous treatment effects. While theory suggests that maximum simulated likelihood estimation is asymptotically consistent, efficient and equivalent to the maximum likelihood estimator when both the number of simulation draws *S* and sample size N $\to \infty$ and $\sqrt{\mathrm{N}}/\ S \to 0$, there is no guidance on how large of an *S* to choose and even theory suggests to experiment. This piece of research reviews strategies of health economists that aim at dealing with this issue. Most pieces of applied research rely on experimentation until numerical stability is achieved, while some employ Monte-Carlo techniques to justify their choice of *S*. A more formal test was suggested, but seemed not to be employed yet. This lack of guidance induces a research problem that needs to be properly addressed.**

I. Introduction

ENDOGENEITY in non-linear regression models arising through self-selection into treatment is a problem very often encountered in, but not limited to, health economics. One prominent situation in which health economists face problems with endogenous regressors is when the effect of health insurance status on healthcare utilisation, such as visits to the doctor, is estimated. Different types of health insurance plans, such as deductibles and co-payments, are offered, to incentivise an economical utilisation of scarce medical resources. As participation in such insurance plans is non-random, selection bias complicates studies in which the effect of endogenous treatment (here: insurance choice) is estimated on a healthcare utilisation outcome, such as number of visits to the doctor. Self-selection occurs when optimising individuals possessing unobservable characteristics, such as awareness of future health states or risk preferences, select health insurance plans accordingly [1]. The same unobservable characteristics that affect insurance choice might then also affect future healthcare utilisation, thus leading to potential unobserved correlation between insurance choice decision and decision to consume health services [2].

One way of addressing this endogeneity issue is the endogenous treatment regression model by [1], that utilises a latent factor structure. Latent factors are incorporated into the treatment and outcome equations, thus allowing to make a distinction between selection on unobservables and selection on observables. As these latent factors cannot be observed, regular maximum likelihood estimation is not feasible. Yet, when assuming a distribution of the latent factors (e.g. standard normal), simulation-based estimation, i.e. *maximum simulated likelihood* (MSL) estimation, remains possible [3].

The properties of MSL estimates crucially depend on the number of simulation draws S (per observation) and sample size N. Given that $S, N \to \infty$ and $\sqrt{N}/S \to 0$, MSL is asymptotically consistent, efficient and equivalent to the maximum likelihood estimator [3]. Yet, this ratio does not provide guidance on what S should be for given N, it only describes the properties of MSL as N increases [4]. Consequently, researchers face a non-trivial problem when deciding how large of an S, given sample size N, to choose. On the one hand, the MSL-approach is computationally burdensome, as it makes extensive use of simulation techniques [5]. Generating random numbers requires a matrix of size $S \times N$, as there are S random draws for each of the N observations. As increasing N will also necessitate an increase in S, this will ultimately lead to non-trivial memory consumption, that is to say, to potentially prohibitively high computational cost [6]. On the other hand, consistency of the estimator requires $S, N \to \infty$ and $\sqrt{N}/S \to 0$. Some [7] recommend using S as large as computational reasonable, while others rely on experimentation with different sizes of S to achieve numerical stability of the estimator as their guide [8], [9], [4], [7]. Thus, the researcher needs to find a suitable trade-off between precision (favouring infinitely large S) on the one hand and computational cost (favouring fixed S) on the other hand. This lack of guidance with respect to choice of an appropriate amount of simulation draws imposes a serious challenge for applied research in two ways. First, having results at hands, the question to the researcher remains, whether or not a sufficient amount of simulation draws was employed [6]. Similarly, and equivalently important, the researcher's choice regarding S remains untraceable to the scientific community.

Consequently, the research problem of the underlying piece of research (work-in-progress) is to find guidance with respect to the choice of an appropriate amount of simulation draws to be employed in maximum simulated likelihood estimation within the endogenous treatment regression context. Establishing such guidance will be beneficial to the research community as it will make MSL-procedure more traceable. As a starting point in establishing such guidance, strategies of dealing with this issue in applied research are presented and discussed within a preliminary literature review. Firstly, however, the MSL-approach and its peculiarities will be explained in more detail, before the relevant literature will be summarised. After the research problem is derived from the literature review, the intended future work to tackle this problem will be discussed.

II. Maximum Simulated Likelihood Estimation

To deal with the endogeneity of treatment (insurance choice) on healthcare utilisation, [1] introduced a latent factor structure into the treatment and outcome equations to account for selection on unobservables. These latent factors enter both treatment and outcome equation to allow for idiosyncratic influences on insurance status choice to affect healthcare utilisation, thus making a distinction between selection on unobservables and selection on observables possible [1]. These latent factors serve as proxies for unobservable characteristics and are interpreted as unobserved heterogeneity. Endogeneity arises, as the same latent factors, i.e. unobservable characteristics, determining insurance choice also affect the healthcare utilisation decision. As they cannot be observed, problems in estimation arise, as no closed-form solution to the respective integral exists [1]. Yet, when making assumptions with respect to the underlying distribution of the unobservable characteristics (e.g. standard normal distribution), maximum simulation likelihood estimation remains feasible. Here, simulation depends on the fact that integrating over a density is simply a form of averaging [10]. Thus, the effect of the unobservable latent factors can be integrated out, resulting in an unbiased (with respect to self-selection) estimate of the treatment effect. Among several possible ways of taking endogeneity into account (e.g. IV-approach, Difference-in-Difference, two-stage residual inclusion) the maximum simulated likelihood-procedure is the only approach that sufficiently addresses both endogeneity of treatment and non-linearity (count data) in the outcome [11].

When outcome y's (e.g. number of doctor visits) conditional density $f(y|\mathbf{x},\ \theta)$, where $\mathbf{x}$ may be individual i's observable characteristics, θ the parameters to be estimated and $\mathbf{u}$ unobservable characteristics, involves such an intractable integral, such that

$$f(y_i|\mathbf{x}_i, \theta) = \int h(y_i|\mathbf{x}_i, \theta, \mathbf{u}_i) g(\mathbf{u}_i) d(\mathbf{u}_i) \qquad (1)$$

requires estimation [6]. Accordingly, one needs to approximate the intractable integral $h(y_i|\mathbf{x}_i, \theta, \mathbf{u}_i)$ with a subsimulator $\tilde{f}(y_i|\mathbf{x}_i, \theta, \mathbf{u}_i^s)$. To do so, S ($S = 1, \ldots, S$) random draws from the assumed distribution of $\mathbf{u}$ are drawn into the subsimulator. The average over S (denoted by $\mathbf{u}_{iS}$) of these subsimulators then provides the simulator $\hat{f}(y_i|\mathbf{x}_i, \theta, \mathbf{u}_{iS})$ such that [6]

$$\underbrace{\hat{f}(y_i|\mathbf{x}_i, \theta, \mathbf{u}_{iS})}_{Simulator} = \frac{1}{S} \sum_{s=1}^{S} \underbrace{\tilde{f}(y_i|\mathbf{x}_i, \theta, \mathbf{u}_i^s)}_{Subsimulator}. \qquad (2)$$

While the usual maximum likelihood estimator maximises the log-likelihood $\ln \mathrm{L}_N(\theta) = \sum_{i=1}^{N} \ln f(y_i|\mathbf{x}_i,\ \theta)$, the maximum simulated likelihood estimator instead maximises the log-likelihood based on the simulated estimation of the density [6]

$$\ln \hat{L}_N(\theta) = \sum_{i=1}^{N} \ln \underbrace{\hat{f}(y_i|\mathbf{x}_i, \theta, \mathbf{u}_{iS})}_{Simulator}. \qquad (3)$$

As the estimator is simulated rather than calculated precisely, simulation error is introduced [10]. This simulation error can be decomposed into three sources of error: simulation chatter, simulation noise and simulation bias [10]. Simulation chatter occurs, when different random draws are used at each likelihood iteration [10], [5]. While simulation chatter might render (simulated) likelihood maximisation infeasible, it can be easily encountered by using the same simulation draws per observation [10], [5]. Thus, simulation chatter does neither depend on the choice of S nor N. Deviations from each simulated value of its expectation lead to simulation noise [10]. As simulation noise cancels out over observations, it decreases with N, even if S is fixed [10]. Simulation bias occurs as the MSL simulator $\ln \hat{f}$ is biased for $\ln f$, even if the simulator $\hat{f}$ is unbiased for f, as a consequence of taking the natural logarithm [6]. An asymptotic bias-adjusted MSL-estimator, that makes use of a bias-adjusted log-likelihood function, is suggested by [3]. As this bias-adjustment assumes bias to be small, [6] adds, that the usefulness of this bias-reduction may vary from case to case, as the small bias-assumption may not always hold. After all, for the simulation bias to disappear, S and $N \to \infty$, while S must increase faster than $\sqrt{N}$, such that $\sqrt{N}/S \to 0$ [3], [10]. If the latter condition is met, MSL is asymptotically normal, efficient and equivalent to maximum likelihood estimation [3], [10]. However, this ratio does not state what S should be for given N, it only describes the properties of the MSL estimator as N increases [4].

III. Literature Review

Whether or not one has done a sufficient amount of simulations to tackle the simulation error issues is a difficult question to answer [6]. As no empirical guidance exists, theory suggest to experiment with different sizes of S until numerical stability of the estimator is achieved [4], [8], [6]. Consequently, [12], [1], [13], [14] report to have relied on such experimentation to find an appropriate S. From experience, [1] suggest, that estimating MSL in the context of endogenous treatment requires "considerably more" simulation draws than models involving seemingly unrelated errors. Yet, [1] do not further elaborate what this might imply in practical terms. Still, [1] state that their choice of S is based on other empirical studies that use MSL. Table I provides a summary of the choice of S, with respect to N, of empirical studies that employed the MSL-approach in the context of endogenous regressors in health economics. While there is no clear guidance on the quantity of simulation draws, consensus seems to exist regarding their quality. Quasi-random draws, such as the Halton-sequence, rather than pseudo-random draws,

TABLE I: OVERVIEW OF CHOICES REGARDING S IN APPLIED HEALTH ECONOMICS RESEARCH.

Reference	N	S	$\frac{\sqrt{N}}{S}$	Random variates
[14]	2,467	1,600	0.031	Halton
[1]	8,129	2,000	0.045	Halton
[12]	26,514	1,000	0.162	Halton
[7]	5,033	400	0.177	Halton
[16]	4,406	300	0.221	Antithetic
[13]	109,349	200	1.653	Halton
[21]	Did not report S			Halton
[11]	Did not report S			Halton
[2]	Did neither report S nor type of random variates			

are considered to greatly reduce the number of simulation draws required for a given amount of precision [10], [4], [5]. Halton-draws are more evenly distributed than pseudo-random draws, while also displaying lower variance, as they are negatively correlated [10]. Even though Halton-draws are rather deterministic than random, [8] add, that when it comes to simulation techniques, the randomness of draws is not as important as their uniform coverage over the domain of integration. Their desirable properties made the Halton-sequence the quasi-random variate of choice, as displayed in Table I. Also, consensus exists that MSL-estimation is, as suggested by theory [4], [6], a rather computationally burdensome approach, as also explicitly stated in several pieces of applied research [13], [12], [7], [15]. Even more so, [12] report to have used less simulation draws than desired (due to having relatively large N) to ensure convergence of their model, while [13] even report that one of their models did not converge after four days of CPU time. One notable deviation of the experimentation-strategy within applied research seems to be the approach by [16] who conducted a Monte-Carlo experiment prior to their empirical study to justify their choice regarding the number of simulation draws. Also, [17] are able to quantify simulation noise and simulation bias of their MSL approach, as their econometric model also offers an analytical solution, to which they can compare their MSL results. Similarly, [18] are able to quantify simulation error, as within their theoretical approach, they employ a simulated dataset, for which the true parameters are known. A different, more formal approach in choosing S is suggested by [19], who describes a diagnostic test, constructed from a Wald test statistic, that captures the magnitude of simulation bias and could be used to compute an amount of S that will produce an acceptable estimator. Even though some pieces of literature [6], [4] point out to this formal test, it was not employed in the reviewed literature. Yet, e.g. [20] employ this diagnostic test in the context of MSL-based dynamic probit models.

IV. DISCUSSION AND OUTLOOK

Within applied research, the question whether one has used enough simulation draws remains challenging. As no clear guidance exists, researchers rely on experimentation with different values of S to achieve numerical stability of the estimator. This procedure does not necessarily satisfy the reader's interest in transparency and traceability with respect to empirical research. One exception [16] in applied health economics research employed a Monte-Carlo study as a benchmark for their subsequent choice of S. Even though translating conclusions drawn from self-designed experimental data to "exogenous" real-world data might similarly raise doubts, it at least seems to be a somewhat more traceable way of justifying one's choice of S. Also, having an analytical, thus correct, solution, as a benchmark, might very much answer the question, whether or not one has used enough simulation draws. Yet, not having an analytical solution remains the motivation to employ MSL in the first place.

This piece of (emerging) research is tackling this overall lack of guidance with respect to choosing S by producing an empirical benchmark within the endogenous treatment context. This benchmark should not be solely based on self-designed experimental data, such as [16], as this type of data might not reflect real-world complexity, that is known to make the MSL-approach burdensome [6], [7]. Nevertheless, such a Monte-Carlo study might clearly be supplemental to reach this overall goal. Also, employing an econometric model on real-world data, for which an analytical solution is possible, does neither seem to be a desirable option, even though the true parameters would be known and could thus serve as a reference. Yet, as already stated, the lack of an analytical solution is the motivation to employ MSL in the first place.

In order to exploit real-world data, while also knowing the true parameter (with respect to selection on unobservables), the Oregon Health Insurance Experiment [22] will be employed. In 2008, within this experiment, a limited amount of Medicaid insurance coverage was allocated randomly to low-income individuals, while also recording healthcare utilisation behaviour of lottery winners and losers afterwards. Randomly assigned Medicaid insurance (i.e. treatment) can be considered exogenous with respect to healthcare utilisation. Thus, employing [1] endogenous treatment regression model, the effect of selection on unobservables on healthcare utilisation will be hypothesised to be zero (due to randomisation). Making use of the Oregon experiment will thus be beneficial to illustrate MSL-convergence behaviour in the context of endogenous treatment regression. These results can ultimately serve as a guide for other health economists to choose an appropriate amount of S, as the simulation error can be estimated quite well, as the true estimates (with respect to selection on unobservables) are known. Thus, MSL-convergence behaviour can be explicitly illustrated for different values of S. Additionally, the formal Wald-based test, suggested by [19], will be employed, to formally support (or reject) the findings. Also, the literature dealing with the MSL-procedure, especially in the realm of health economics, will be reviewed more extensively and intensively, with respect to strategies of choosing an appropriate S, while also promising alternatives to the MSL-approach, as suggested by [18], need to be studied closely. As a result, health econometricians will benefit from this ongoing piece of research, as it will provide them with some guidance whether or not they have chosen a sufficiently large S, when employing MSL estimation.

V. References

[1] P. Deb and P. K. Trivedi. “Specification and simulated likelihood estimation of a non–normal treatment–outcome model with selection: Application to health care utilization”. In: *The Econometrics Journal* 9.2 (2006), pp. 307–331. DOI: 10.1111/j.1368-423X.2006.00187.x.

[2] D. Shane and P. K. Trivedi. “What drives differences in health care demand? The role of health insurance and selection bias”. In: *Health, Econometrics and Data Group Working Papers* 12/09 (2012).

[3] C. Gouriéroux and A. Monfort. *Simulation-based econometric methods*. Oxford University Press, 1997. DOI: 10.1093/0198774753.001.0001.

[4] W. H. Greene. *Econometric analysis*. 6. ed. Upper Saddle River, NJ: Pearson Prentice Hall, 2008.

[5] A. Cameron and P. K. Trivedi. *Microeconometrics using Stata*. Rev. ed. A Stata Press publication. College Station, Tex.: Stata Press, 2010.

[6] A. Cameron and P. K. Trivedi. *Microeconometrics: Methods and applications*. New York, NY: Cambridge University Press, 2005.

[7] P. Deb and P. K. Trivedi. “Maximum simulated likelihood estimation of a negative binomial regression model with multinomial endogenous treatment”. In: *Stata Journal* 6.2 (2006), pp. 246–255.

[8] D. M. Drukker and R. Gates. “Generating Halton sequences using Mata”. In: *Stata Journal* 6.2 (2006), 214–228(15).

[9] D. M. Drukker. “Maximum simulated likelihood: Introduction to a special issue”. In: *Stata Journal* 6.2 (2006), 153–155(3).

[10] K. E. Train. *Discrete choice methods with simulation*. Cambridge: Cambridge University Press, 2009. DOI: 10.1017/CBO9780511805271.

[11] M. M. Garrido, P. Deb, J. F. Burgess, and J. D. Penrod. “Choosing models for health care cost analyses: issues of nonlinearity and endogeneity”. In: *Health services research* 47.6 (2012), pp. 2377–2397. DOI: 10.1111/j.1475-6773.2012.01414.x.

[12] P. Deb, C. Li, P. K. Trivedi, and D. M. Zimmer. “The effect of managed care on use of health care services: results from two contemporaneous household surveys”. In: *Health economics* 15.7 (2006), pp. 743–760. DOI: 10.1002/hec.1096.

[13] V. Atella and P. Deb. “Are primary care physicians, public and private sector specialists substitutes or complements? Evidence from a simultaneous equations model for count data”. In: *Journal of health economics* 27.3 (2008), pp. 770–785. DOI: 10.1016/j.jhealeco.2007.10.006.

[14] M. Bratti and A. Miranda. “Endogenous treatment effects for count data models with endogenous participation or sample selection”. In: *Health economics* 20.9 (2011), pp. 1090–1109. DOI: 10.1002/hec.1764.

[15] A. Geraci, D. Fabbri, and C. Monfardini. “Testing exogeneity of multinomial regressors in count data models: Does two-stage residual inclusion work?” In: *Journal of Econometric Methods* 7.1 (2018), p. 313. DOI: 10.1515/jem-2014-0019.

[16] M. K. Munkin and P. K. Trivedi. “Simulated maximum likelihood estimation of multivariate mixed–Poisson regression models, with application”. In: *The Econometrics Journal* 2.1 (1999), pp. 29–48. DOI: 10.1111/1368-423X.00019.

[17] V. Atella and A. Holly. “Disentangling adverse selection, moral hazard and supply induced demand: An empirical analysis of the demand for healthcare services”. In: *SSRN Electronic Journal* (2016). DOI: 10.2139/ssrn.2801679.

[18] C. R. Bhat, C. Varin, and N. Ferdous. “A comparison of the maximum simulated likelihood and composite marginal likelihood estimation approaches in the context of the multivariate ordered-response model”. In: *Maximum simulated likelihood methods and applications*. Ed. by R. C. Hill and W. H. Greene. Vol. 26. Advances in Econometrics. Bingley, UK: Emerald, 2010, pp. 65–106. DOI: 10.1108/S0731-9053(2010)0000026007.

[19] V. A. Hajivassiliou. “Some practical issues in maximum simulated likelihood”. In: *Simulation-based Inference in Econometrics*. Ed. by R. Mariano, T. Schuermann, and M. J. Weeks. Cambridge: Cambridge University Press, 2000, pp. 71–99.

[20] P. Contoyannis, A. M. Jones, and N. Rice. “Simulation-based inference in dynamic panel probit models: An application to health”. In: *Empirical Economics* 29.1 (2004), pp. 49–77. DOI: 10.1007/s00181-003-0189-x.

[21] M. B. Buntin, C. H. Colla, P. Deb, N. Sood, and J. J. Escarce. “Medicare spending and outcomes after postacute care for stroke and hip fracture”. In: *Medical care* 48.9 (2010), pp. 776–784. DOI: 10.1097/MLR.0b013e3181e359df.

[22] A. Finkelstein et al. *The Oregon health insurance experiment: Evidence from the first year*. Cambridge, MA, 2011. DOI: 10.3386/w17190.

25th Conference on Knowledge Acquisition and Management

KNOWLEDGE management is a large multidisciplinary field having its roots in Management and Artificial Intelligence. Activity of an extended organization should be supported by an organized and optimized flow of knowledge to effectively help all participants in their work.

We have the pleasure to invite you to contribute to and to participate in the conference “Knowledge Acquisition and Management". The predecessor of the KAM conference has been organized for the first time in 1992, as a venue for scientists and practitioners to address different aspects of usage of advanced information technologies in management, with focus on intelligent techniques and knowledge management. In 2003 the conference changed somewhat its focus and was organized for the first under its current name. Furthermore, the KAM conference became an international event, with participants from around the world. In 2012 we’ve joined to Federated Conference on Computer Science and Systems becoming one of the oldest event.

The aim of this event is to create possibility of presenting and discussing approaches, techniques and tools in the knowledge acquisition and other knowledge management areas with focus on contribution of artificial intelligence for improvement of human-machine intelligence and face the challenges of this century. We expect that the conference&workshop will enable exchange of information and experiences, and delve into current trends of methodological, technological and implementation aspects of knowledge management processes.

Topics

- Knowledge discovery from databases and data warehouses
- Methods and tools for knowledge acquisition
- New emerging technologies for management
- Organizing the knowledge centers and knowledge distribution
- Knowledge creation and validation
- Knowledge dynamics and machine learning
- Distance learning and knowledge sharing
- Knowledge representation models
- Management of enterprise knowledge versus personal knowledge
- Knowledge managers and workers
- Knowledge coaching and diffusion
- Knowledge engineering and software engineering
- Managerial knowledge evolution with focus on managing of best practice and cooperative activities
- Knowledge grid and social networks
- Knowledge management for design, innovation and eco-innovation process
- Business Intelligence environment for supporting knowledge management
- Knowledge management in virtual advisors and training
- Management of the innovation and eco-innovation process
- Human-machine interfaces and knowledge visualization

Event Chairs

- **Hauke, Krzysztof,** Wroclaw University of Economics, Poland
- **Nycz, Malgorzata,** Wroclaw University of Economics, Poland
- **Owoc, Mieczyslaw,** Wroclaw University of Economics, Poland
- **Pondel, Maciej,** Wroclaw University of Economics, Poland

Program Committee

- **Abramowicz, Witold,** Poznan University of Economics, Poland
- **Andres, Frederic,** National Institute of Informatics, Tokyo, Japan
- **Bodyanskiy, Yevgeniy,** Kharkiv National University of Radio Electronics, Ukraine
- **Chmielarz, Witold,** Warsaw University, Poland
- **Christozov, Dimitar,** American University in Bulgaria, Bulgaria
- **Jan, Vanthienen,** Katholike Universiteit Leuven, Belgium
- **Mercier-Laurent, Eunika,** University Jean Moulin Lyon3, France
- **Sobińska, Małgorzata,** Wroclaw University of Economics, Poland
- **Surma, Jerzy,** Warsaw School of Economics, Poland and University of Massachusetts Lowell, United States
- **Vasiliev, Julian,** University of Economics in Varna, Bulgaria
- **Zhu, Yungang,** College of Computer Science and Technology, Jilin University, China

Organizing Committee

- **Hołowińska, Katarzyna**
- **Przysucha, Łukasz,** Wroclaw University of Economics

DOI: 10.15439/2019F357
ISSN 2300-5963 ACSIS, Vol. 19

Automation of signing contracts for learning in educational units

Agnieszka Sawicka
Wrocław University of Economics
ul. Komandorska 118/120 53-345
Wrocław, Poland
Email:agnieszka.sawicka@ue.wroc.pl

Abstract—**Non-public institutions play an important role in higher education. The article discusses the development of administrative repairs through RPA. Based on the robotic process automation, a dedicated project was presented to the automation.**

I. Introduction

TECHNICAL and technological innovations are to contribute to creating better solutions that help in automating administrative processes taking place in educational units. Important in creating innovation and the adaptation of existing solutions has not practiced in the field of higher education, the need is very detailed analysis of the organization and the processes taking place in it. The latest technologies are a challenge for today's universities. They can help create creative solutions that will contribute to the development of the organization, affect the development of the individual and help maintain high quality education. The solution proposed in the article aims to relieve the recruitment staff and the dean's offices. It has to minimize the procedures related to the circulation of documentation, including the signing of an agreement on science and issues of storage. The proposed functionality will also significantly affect the improvement of ecology in the university environment, which is not without significance for more and more aware students in this matter. The project has been prepared based on the operation of a non-public entity, however you can be so universal that it can also be used by public universities - the larger and smaller.

II. The role of RPA technology and ERP systems In supporting the circulation of documentation in the education industry

Investments in new, intelligent technologies bring development and build the future of innovation in many enterprises both in Poland and on the global market. They also offer specific benefits, such as reducing costs and obtaining more useful data and information. On the subject of RPA technology (Robotic Process Automation) , which enters the administration , with the aim of automating repetitive processes, you can find many interesting publications, but the topic of RPA could also be used to manage the circulation of documentation in a private university. Just as in the case of ERP systems that support the circulation of documentation in the education industry, the RPA technology is designed to automate and replace the repetitive , measurable activities that have been performed so far by man. Thus, it reduces human error and improves the functioning of the entire organization. Replacement works human operator, who celebrates part, cap and processing data, and sentence her to be an idea and solution to the issue of reorganization and documentation university. In the following will be considered a case of signing contracts for storage and science, however, these technologies could also help s to work Bursary department, human resources or the rector of the university and shapes - not just the public. This technology is a element of intellectual above process automation (IPA - intelligent process automation), which means that the used solutions are based on logic circuits that perform pre-programmed operations, benefiting from the services about data, which further seems to confirm the legitimacy of considering the introduction of smart technologies to the modern administration of a non-public university.

III. Dedicated solutions for reorganizing documentation for a non-public university

As already mentioned, the RPA technology and ERP class systems are designed to support processes and take repetitive tasks. They should also eliminate the risk of clerical errors and improper storage of the most important documents for the university, which is undoubtedly the education contract. In non-public universities, the contract regulates the payment issues and is a set of rules and procedures in specific cases. Therefore, an attempt was made to design a solution based on intelligent technologies, which aims at signing learning contracts by accepting the terms of the contract and storing it in electronic form, limiting highly repetitive activities for the employees of the dean's offices or recruitment departments.

The organization within the scope of the documentation kept, which from 1 October 2019 will be allowed by the Ordinance of the Minister of Science and Higher Education, among others:

- student ID cards will be issued only in electronic form,
- the hologram of the ID card placed in the fields marked in succession will become a print of strict accountancy,
- reviews of diploma theses will become public,
- in principle, changes in the required elements on the diploma of graduation,
- resignation from issuing decisions on admission to university,
- storing student documents in the so-called electronic student file .

The electronic student file is intended to limit the storage of a large number of documents and to minimize the scope of archiving work at universities . Reducing the number of printing, including paper and toners, will undoubtedly improve the ecology of the university and reduce its costs. Electronic briefcase student carries with it the possibility of storage contracts for science in the work of organizing the deans' offices. Using intelligent technologies, instead of storing a given print in the form of scans of signed contracts for learning in a formal briefcase, you can expand the functionality of the system supporting the dean's office work and extend it with a module that would allow you to sign a learning contract in electronic form. Such a document would go straight to the electronic portfolio of the selected student, thus limiting the scope of procedures and the repetitive work done by the employees of the recruitment departments or dean's offices - in line with the idea of RPA . This solution is admissible under Article 60 of the Civil Code, in which "(...) the will of the person making a legal transaction may be expressed by any behavior of that person that reveals his will in a sufficient manner, including by revealing that will in electronic form ".

The signing of the contract would take place in the following steps:

- to the e-mail address provided in the recruitment form there is a link to the page with a learning contract asking for familiarization with its content,
- after learning the terms of the contract, the student is asked to accept it through the "I accept the terms of the learning contract" button,
- if you select this be an agreement concluded in electronic form to e legal consequences resulting from its signing .

After the student has accepted the provisions , the confirmation along with the contract is sent automatically in the .pdf file to the new system contracting module and the dean's office will be notified and the information about the acceptance of the new contract is sent via e-mail to the mailbox of the indicated department employee , who accepts the contract to the system after acceptance . For universities that do not use the electronic portfolio tool, it would be possible to store files of this type in the cloud or on servers. Dedicated extension of the selected non-public university, he would also allow for printing such an agreement, the importance of which would be tantamount to the agreement signed in the traditional way and stored in a standard student carried a briefcase act. The described process is presented in the figures below.

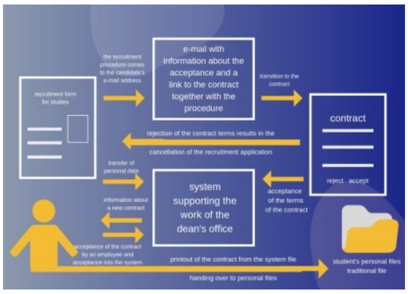

recruitment form
for studies
the recruitment procedure comes to the candidate's e-mail address
e-mail with information about the acceptance and a link to the contract together with the procedure
transition to the contract
contract
reject accept
rejection of the contract terms results in the
cancellation of the recruitment application
transfer of personal data
system supporting the work of the dean's office
acceptance of the terms of the contract
information about a new contract
acceptance of the contract by an employee and acceptance into the system
printout of the contract from the system file
handing over to personal files
student's personal files traditional file

Figure 1: Scheme of electronic procedures for submitting science agreements

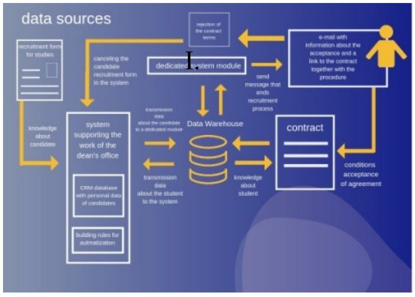

data sources
recruitment form for studies
rejection of the contract terms
canceling the candidate recruitment form in the system
dedicated system module
e-mail with information about the acceptance and a link to the contract together with the procedure
send message that ends recruitment process
knowledge about candidate
system supporting the work of the dean's office
CRM database with personal data of candidates
building rules for autmatization
transmission data about the candidate to a dedicated module
Data Warehouse
transmission data about the student to the system
knowledge about student
contract
conditions acceptance of agreement

Figure 2: Scheme of data transfer acquiring knowledge between the system and the user

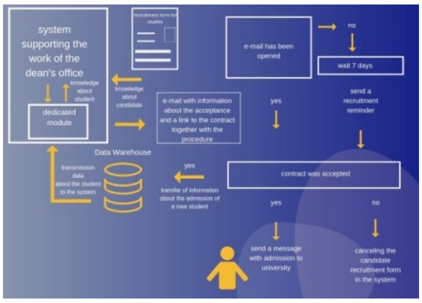

system supporting the work of the dean's office
knowledge about student
dedicated module
recruitment form for studies
knowledge about candidate
e-mail has been opened
no
wait 7 days
e-mail with information about the acceptance and a link to the contract together with the procedure
yes
send a recruitment reminder
Data Warehouse
transmission data about the student to the system
yes
transfer of information about the admission of a new student
contract was accepted
yes
no
send a message with admission to university
canceling the candidate recruitment form in the system

Figure 3: Scheme of automated transfer of information about the contract and admission between the university and candidate

The above diagrams (Fig. 2 and Fig. 3) present it operation of a dedicated solution and its connection with the system together with the automation of messages sent to candidates. These solutions have been designed based on technologies used to build autoresponders . The mechanism introduced into the dedicated contract module, which will be responsible for the preparation of data collected and collected in the data warehouse, will operate on the basis of a group of processes. The following procedures are used to automate the repetitive and tedious processes performed by employees:

- notification of a new recruitment form in the database of the system supporting the university,

- analysis of the application in terms of missing data,
- error analysis in the provided e-mail addresses and feedback on the incorrect form to the employee,
- determining the potential number of messages with contracts to send,
- automatic sending of messages notifying many candidates at the same time about the acceptance of the application, completion of the recruitment process and the possibility of signing the contrach,
- analysis of the number of sent messages and the number of returned, signed contracts,
- re-sending messages reminding you that you have completed the formalities after passing the specified period,
- recording of reported non-standard cases,
- activities supporting and analyzing the effectiveness of the process.

IV. Advantages of the prepared solution for a private university

In the abovementioned , which became an example to creation of the above mechanism, one of the most important advantages is the simplification of admission procedures for the new student. Dean's office employees are responsible for all types of studies and courses , so the obligation to accept recruitment documents falls on the dean's office . Due to the large number of duties, automation of recurring processes will significantly improve the efficiency of the department, reduce costs and streamline the admission process. The bigger the university, the more benefits it can take from the proposed solution , because the contract module can be freely expanded by the necessary functionalities or create similar solutions dedicated only to the specific needs of organizational units . It is worth noting that filling out contract forms, sending e-mail with identical content and identical links are tasks that can be automated without any problems, thus gaining time for duties that require more attention.

V. Conclusion

Intelligent technologies undoubtedly bring access both in the field of science and in the socio-economic environment. It is therefore recommended that public and non-public higher education institutions should use solutions that are supported by the latest technological solutions, in order to improve the quality of education, minimize errors in the circulation of administrative documentation and the course of study. Each solution that carries the features of innovation gives the opportunity to meet the expectations of future candidates, thus building the prestige of the university and positively influences the reception of the individual. The introduction of automation of repetitive activities in the administrative departments of universities is a new approach, which may raise some kind of controversy . However, broadening the horizons in the field of modern technologies and introducing its functionalities to departments and organizational units, in which repetitive activities are on the agenda, is caused by the constant need for change, the need to keep up with the changing requirements. The search for creative solutions is the equation between the known, acceptable order and the chaos that change can bring, innovation. Innovation, its implementation and constant search for new solutions, treating it as a direction of action, and as a way of management can dispel fears of disturbing relative stability.

References

[1] Chmielarz W., “Project management and development of management information systems”, University of Warsaw WZ Publishing House, Warsaw 2013.

[2] Forbes Technology Council, “10 Strategic Ways To Automate Your Internal Business Workflows”, https://www.forbes.com/sites/forbestechcouncil/2018/05/29/10-strategic-ways-to-automate-your-internal-business-workflows/# 2dfd6e536972 , (accessed: 06/06/2019).

[3] “How Workflow Systems and Automation Are Transforming Life at the Office”, https://www.business.com/articles/workflow-automation-transforming-office/ , (access : 06/06/2019).

[4] “How Colleges Can Use Workflow Automation to Cut Costs”, https://gravityflow.io/articles/how-colleges-can-use-workflow-automation-to-cut-costs/ , (accessed: 06/06/2019).

[5] Economic IT, “Computerization of the economic circulation”, ed. Korczak J., Dyczkowski M., Łukasik-Makowska B., Publisher of University of Economics in Wroclaw, Wroclaw 2013..

[6] Januszewski A., “Functionality of IT management systems t1.t2.”, PWN Scientific Publishers, Warsaw 2006.

[7] “The electronic economy system in the information and knowledge age”, ed. Olszak C., Ziemba E., PWN Scientific Publisher, Warsaw 2006.

[8] White Stephen A., “Introduction to BPMN”, IBM Corporation, (accessed: 06/06/2019).

[9] Wójtowicz R., “Selected legal, organizational and technological aspects of archiving electronic documents in Polish public institutions, Acquisition of knowledge and knowledge management”, Publisher of University of Economics in Wroclaw, Wroclaw 2010.

6th Doctoral Symposium on Recent Advances in Information Technology

THE aim of this meeting is to provide a platform for exchange of ideas between early-stage researchers, in Computer Science and Information Systems, PhD students in particular. Furthermore, the symposium will provide all participants an opportunity to get feedback on their studies from experienced members of the IT research community invited to chair all DS-RAIT thematic sessions. Therefore, submission of research proposals with limited preliminary results is strongly encouraged.

Besides receiving specific advice for their contributions all participants will be invited to attend plenary lectures on conducting high-quality research studies, excellence in scientific writing and issues related to intellectual property in IT research. Authors of the two most outstanding submissions will have a possibility to present their papers in a form of short plenary lecture.

TOPICS

- Automatic Control and Robotics
- Bioinformatics
- Cloud, GPU and Parallel Computing
- Cognitive Science
- Computer Networks
- Computational Intelligence
- Cryptography
- Data Mining and Data Visualization
- Database Management Systems
- Expert Systems
- Image Processing and Computer Animation
- Information Theory
- Machine Learning
- Natural Language Processing
- Numerical Analysis
- Operating Systems
- Pattern Recognition
- Scientific Computing
- Software Engineering

EVENT CHAIRS

- **Kowalski, Piotr,** Systems Research Institute, Polish Academy of Sciences; AGH University of Science and Technology, Poland
- **Łukasik, Szymon,** Systems Research Institute, Polish Academy of Sciences, AGH University of Science and Technology, Poland

PROGRAM COMMITTEE

- **Arabas, Jaroslaw,** Warsaw University of Technology, Poland
- **Atanassov, Krassimir T.,** Bulgarian Academy of Sciences, Bulgaria
- **Balazs, Krisztian,** Budapest University of Technology and Economics, Hungary
- **Bronselaer, Antoon,** Department of Telecommunications and Information at Ghent University, Belgium
- **Castrillon-Santana, Modesto,** University of Las Palmas de Gran Canaria, Spain
- **Charytanowicz, Malgorzata,** Catholic University of Lublin, Poland
- **Corpetti, Thomas,** University of Rennes, France
- **Courty, Nicolas,** University of Bretagne Sud, France
- **De Tré, Guy,** Faculty of Engineering and Architecture at Ghent University, Belgium
- **Fonseca, José Manuel,** UNINOVA, Portugal
- **Fournier-Viger, Philippe,** University of Moncton, Canada
- **Gil, David,** University of Alicante, Spain
- **Herrera Viedma, Enrique,** University of Granada, Spain
- **Hu, Bao-Gang,** Institute of Automation, Chinese Academy of Sciences, China
- **Koczy, Laszlo,** Szechenyi Istvan University, Hungary
- **Kokosinski, Zbigniew,** Cracow University of Technology, Poland
- **Krawiec, Krzysztof,** Poznan University of Technology, Poland
- **Kulczycki, Piotr,** Systems Research Institute, Polish Academy of Sciences, Poland
- **Kusy, Maciej,** Rzeszow University of Technology, Poland
- **Lilik, Ferenc,** Szechenyi Istvan University, Hungary
- **Lovassy, Rita,** Obuda University, Hungary
- **Malecki, Piotr,** Institute of Nuclear Physics PAN, Poland
- **Mesiar, Radko,** Slovak University of Technology, Slovakia
- **Mora, André Damas,** UNINOVA, Portugal
- **Noguera i Clofent, Carles,** Institute of Information Theory and Automation (UTIA), Academy of Sciences of the Czech Republic, Czech Republic
- **Pamin, Jerzy,** Institute for Computational Civil Engineering, Cracow University of Technology, Poland
- **Petrik, Milan,** Czech University of Life Sciences Prague, Faculty of Engineering, Department of Mathematics, Czech Republic
- **Ribeiro, Rita A.,** UNINOVA, Portugal

- **Sachenko, Anatoly,** Ternopil State Economic University, Ukraine
- **Samotyy, Volodymyr,** Lviv State University of Life Safety, Ukraine
- **Szafran, Bartlomiej,** Faculty of Physics and Applied Computer Science, AGH University of Science and Technology, Poland
- **Tormasi, Alex,** Szechenyi Istvan University, Hungary
- **Wei, Wei,** School of Computer science and engineering, Xi'an University of Technology, China
- **Wysocki, Marian,** Rzeszow University of Technology, Poland
- **Yang, Yujiu,** Tsinghua University, China
- **Zadrozny, Slawomir,** Systems Research Institute, Poland
- **Zajac, Mieczyslaw,** Cracow University of Technology, Poland

Position Papers of the Federated Conference on Computer Science and Information Systems pp. 91–96
DOI: 10.15439/2019F345
ISSN 2300-5963 ACSIS, Vol. 19

Integrating Computer Vision and Natural Language Processing to Guide Blind Movements

Lenard Nkalubo
Department of Computer Science
Kyambogo University

Abstract—**Vision is the most essential sense for human beings. Vision impairment is one of the most problems faced by the elderly. Blindness is a state of lacking the visual perception due to physiological or neurological factors. This paper presents a detailed systematic and critical review that explores the available literature and outlines the research efforts that have been made in relation to movements of the blind and proposes an integrated guidance system involving computer vision and natural language processing. An advanced Smartphone equipped with vision, language and intelligence capabilities is attached to the blind person in order to capture surrounding images and is then connected to a central server programmed with a faster region convolutional neural network algorithm and an image detection algorithm to recognize images and multiple obstacles. The server sends the results back to the Smartphone which are then converted into speech for the blind person's guidance.**

Index Terms—**Computer vision, Smartphone-based, Faster CNN algorithm, visually impaired, natural language processing.**

I. Introduction

TRADITIONALLY, movements of the blind are guided by a walking stick. As technologies improve, smart walking sticks have been explored by embedding sensors on the walking sticks. Other attempts have also been tried with the use of electronic travel aids [1], electronic orientation aids (EOAs) [13] and position locator devices (PLDs) [10]. Despite all the efforts undertaken to solve the movement of the blind, it remains challenging and requires more research endeavors [2].

This paper gives the state of the art and outlines the research efforts in relation to the techniques involved in the movement of the blind.

The rest of this paper is organized as follows: section 2 discusses the research motivation or concern, section 3 provides the current literature about the techniques involved in solving the problem, section 4 gives the methodology, section 5 gives the intended research product and section 6 gives the conclusion.

II. Research Motivation

Globally, it is estimated that approximately 1.3 billion people live with some form of distance or near vision impairment. With regards to distance vision, 188.5 million have mild vision impairment, 217 million have moderate to severe vision impairment, and 36 million people are blind [3]. With regards to near vision, 826 million people live with a near vision impairment. The study carried out in [3] identified 288 studies of 3,983,541 participants contributing data from 98 countries. Among the global population with moderate or severe vision impairment in 2015 (216·6 million [80% uncertainty interval 98·5 million to 359·1 million]), the leading causes were uncorrected refractive error (116·3 million [49·4 million to 202·1 million]), cataract (52·6 million [18·2 million to 109·6 million]), age-related macular degeneration (8·4 million [0·9 million to 29·5 million]), glaucoma (4·0 million [0·6 million to 13·3 million]), and diabetic retinopathy (2·6 million [0·2 million to 9·9 million]) [3].

Furthermore, 81 percent of people with vision impairment are aged 50 and above years. Apart from age, other causes of vision impairment have been found to be cataracts, glaucoma, diabetic retinopathy, and uncorrected refractive errors [6]. The number of people affected by the common causes of vision loss has increased substantially as the population increases and ages. Preventable vision loss due to cataract (reversible with surgery) and refractive error (reversible with spectacle correction) continue to cause most cases of blindness and moderate or severe vision impairment in adults aged 50 years and older. A large scale-up of eye care provision to cope with the increasing numbers is needed to address avoidable vision loss. [3]

Arising from this statistics, it is clear that the problem of vision impairment cannot be addressed fully from the medical perspective. We have to explore other alternatives that support those already in existance since blindness or vision impairment is function of age which puts the aging persons at high risk of becoming visually impaired.

A. Research Objective

The general objective of the paper is to identify current literature, current research efforts in solving the problem of the movements for the blind or unpaired people while proposing the latest solution which emerges from current technological advancements in artificial intelligence with the integration of computer vision and NLP. In a more specific way this would be done by finding out the current strength and weakness of the blind movement solutions, identifying computer vision Algorithms and NLP capabilities especially deep learning CNN algorithms and latest smart phones supporting natural language capabilities.

B. Research Question

The general research question would be to find out how can the current research efforts of solving the vision impairment problem be analyzed? How can new trends in technology like computer vision and Natural language processing be used in solving the problem?

In more specific terms the research questions would address: What are the strengths and weaknesses of the current blind movement solution systems? What are the requirements of the integrated computer vision and Natural language processing proposed solution? How can a faster CNN-algorithm be implemented with language capabilities to provide a solution for blind movements?

In summary, according to the implementations in previous studies, assistive devices for navigation for visually impaired people still focus on location and distance sensing and alerting users on the types of obstacles in front of them and their surroundings. Therefore, the practicability of such assistive devices is very low due to the cost and vulnerability to damage from the sun and rain [9]. Therefore, this paper addresses those obstacles and proposes a mobile navigation system for visually impaired people; this system employs an advanced Smartphone and with deep learning algorithms to recognize various obstacles and is not limited to indoor or outdoor environments.

The Google 3xL Smartphone released in November 2018 whose Image processor and sensor is shown in (Fig 1). is equipped with machine learning capabilities like text-speech, image recognition, voice processing and facial recognition and has Google global positioning system features (A-GPS, GLONASS, BDS, and GALILEO). The Smartphone has Google applications like the Google's cloud, maps, lens, and assistant. This latest advanced Smartphone addresses gaps existing in the old non mobile systems and makes the study address the research gap in previous implementations. The study, investigation, analysis, design and implementation of these new technologies will squarely bridge the research gap and contribute a new knowledge base towards bridging computer vision and natural language processing.

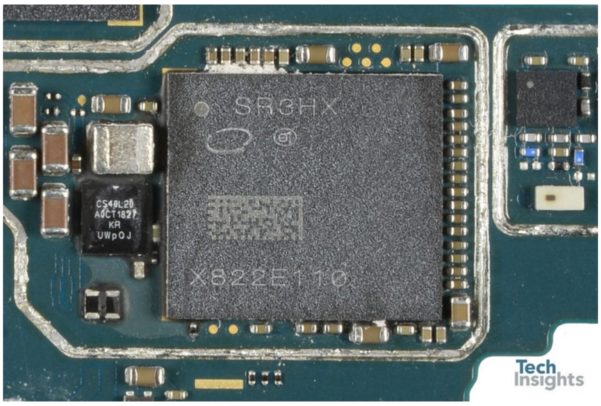


Fig. 1. Google – Pixel - 3XL –Image Processor and Sensor

III. Literature Review

A. Existing Technologies

A smart blind guidance device has been proposed by [1] which uses infrared sensors and includes a small hand stick along with a wearable head set. Another system known as Sonic Path Finder shown in (Fig 2). by [4] which works based on ultrasound transmission and reception but it is not a hand held system, it is attached on the head of the user but is unable to provide the accurate path and the position of an obstacle. It is a secondary mobility aid for use by people with vision impairment. It is not suitable for anyone who does not have primary mobility skills. It is designed for use out-of-door in conjunction with a cane, guide dog or residual vision [4].

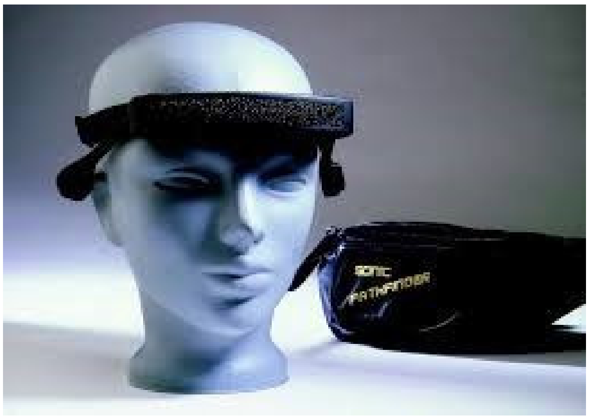


Fig. 2. Sonic Path Finder

ETAs are general assistant devices to help visually impaired people avoid obstacles [5]. Microsoft Kinect shown in (Fig. 3) is usually used as the main recognition hardware in such systems [11]. However, Microsoft Kinect cannot be used in environments with strong light. Moreover, it can determine only the presence of obstacles ahead [8] or recognize a few types of obstacles in few related studies [7]. In general camera recognition systems are designed to recognize tactile or obstacle images.

Fig. 3. Microsoft Kinetic Sensor

A combination of a camera and other multiple sensors is usually used to get more information to draw the shapes of passageway and obstacles [7]. Thus these systems may provide a guiding service and a recognition result out of a few types of obstacles. The drawback of EOAs is that they need more complex computing to hardly be realized as a real-time and lightweight guiding device. PLDs are used to determine the precise position of its holder such as devices that use global positioning system (GPS) and geographic information system (GIS) technologies [2].

B. Computer Vision

Computer Vision (CV) tasks can be summarized by the concept of 3Rs [12], which are reconstruction, recognition, and reorganization. Reconstruction involves estimating the three-dimensional (3D) scene that gave rise to a particular visual image. This representation is shown in (Fig 4). It can be accomplished using a variety of processes incorporating information from multiple views, shading, texture, or direct depth sensors. Reconstruction process results in a 3D model, such as point clouds or depth images. Some examples for reconstruction tasks are Structure from Motion, scene reconstruction, and shape from shading. Recognition involves both 2D problems (like handwritten recognition, face recognition, scene recognition, or object recognition), and 3D problems (like 3D object recognition from point clouds which assists in robotics manipulation). Recognition results in assigning labels to objects in the image. Reorganization involves bottom-up vision: segmentation of the raw pixels into groups that represent the structure of the image. Reorganization tasks range from low-level vision like edge, contour, and corner detection, intrinsic images, and texture segmentation to high-level tasks like semantic segmentation [15], which has an overlapping contribution to recognition tasks. A scene can be segmented based on low-level vision] or high-level information like shadow segmentation that utilizes class information.

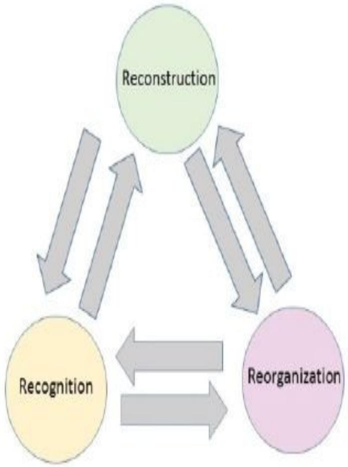


Fig.4 The 3Rs in computer vision (Malik et al. 2016)[12], which are reconstruction, reorganization and recognition.

C. Natural Language Processing (NLP)

Following the Vauquois triangle for machine translation shown in (Fig 5).[17], Natural Language Processing (NLP) tasks can be summarized into concepts ranging from syntax to semantics and to pragmatics at the top level to achieve communication. Syntax includes morphology (the study of word forms) and compositionality (the composition of smaller language units like words to larger units like phrases or sentences). Semantics is the study of meaning, including finding relations between words, phrases, sentences or discourse. Pragmatics studies how meaning changes in the presence of a specific context. For instance, an ironic sentence cannot be correctly interpreted without any side information that indicates the indirectness in the speaker's intention.

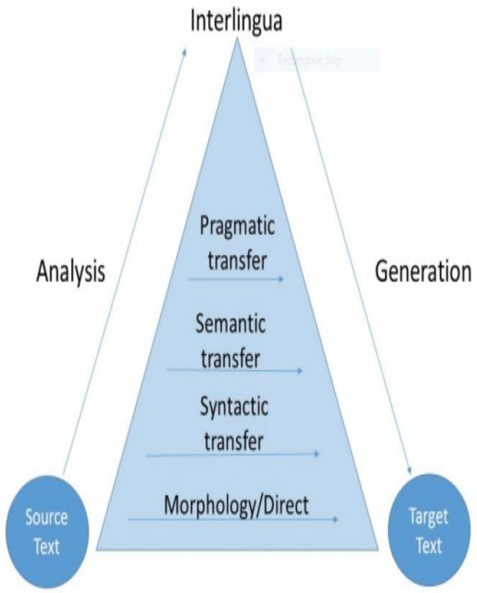


Fig 5. The Vauquois triangle for machine translation (Vauquois, 1968) [17].

Ambiguity in language interpretation a main obstacle for an intelligent system to overcome and achieve language understanding. Some complex tasks in natural language processing include machine translation, information extraction, dialog interface, question answering, parsing, and summarization. There is always meaning lost when translating between one language and another. When "translating" between the low-level pixels or contours of an image and a high level description in word labels or sentences, there is a wide chasm to be crossed. Bridging the Semantic gap means building a bridge from visual data to language data like words or phrases.

D. Conceptual Framework

As a general framework shown in (Fig. 6), most methods in image captioning are trying to either model language information as another layer on top or jointly model language and vision simultaneously by a carefully designed loss function or algorithm. These systems consider structural multimodal input and create structural output in contrast to the traditional system.

[15] Unifies language and vision for robotics again by bridging visual, language, speech, and control data for a forklift robot. Their robot can recognize objects based on one example using one-shot visual memory. Its natural language interface works by speech processing or pen gestures.

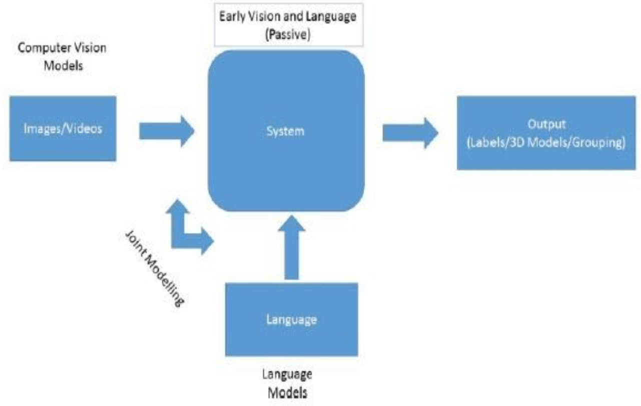


Fig.6. The early vision and language conceptual framework which passively utilizes high-level language information as an additional context. memory.

[18] provides a survey for verbal and nonverbal human-robot interaction.

A video can be described by a sentence or a discourse that is a structured set of sentences that tells a specific story. A sentence prior can be learned from web-scale corpora to bias the model and penalize unlikely combinations of actors/actions/objects [19]. It links the compositional structure of languages and the compositional structure of video events using natural language semantics and three essential computer vision tasks which are tracking, object detection, and event recognition. These three tasks are done simultaneously using a single cost function including the attention mechanism to focus on the most salient event to produce the best sentence description for activity recognition.

From a natural language processing perspective, the sentence tracker utilizes lexical semantics and contains the information of “who did what to whom, and where and how they did it” [20]. An object is described as a noun phrase; the observed action is described as a verb; object properties are described as adjectives; spatial relations between objects are described by prepositions; and the event characteristics are represented as prepositions and adverbs. The system has a predefined vocabulary and a sentence is composed using a set of predefined grammars.

The sentence tracker can be divided into two subsystems [20]. The first subsystem consists of three steps. First, object detection is performed with a high-recall setting. Second, tracking with forward projection to increase precision is performed. Third, the optimal set of detections is chosen using dynamic programming with the Viterbi algorithm, outputting a result consistent with the optical flow. For the second subsystem, events are recognized using HMMs, also computed using the Viterbi algorithm. The unified objective function from the final step of the first subsystem and the second subsystem can be merged, since both are based on HMMs.

Recent approaches deploy powerful deep-learning frameworks to model both image and word sequences. These approaches can support a larger vocabulary than other methods that have a small set of predefined vocabulary [21]. Similar to image captioning, [21] combines a sequence of CNN and another two sequences of LSTM to generate sentence description from video. AlexNet is deployed as a pretrained CNN model, and the output features are mean-pooled before feeding to the LSTM sentence decoding layer. This work is inspired by [22], which uses CRF to extract image features for the intermediate representation for an LSTM. [21] makes an improvement to [22] that discards the temporal information and models the image frames as a bag-of-images.

E. Fast CNN (Deep Learning)

There are many attempts in many benchmarks in open competitions to design a better architecture of CNNs. Some notable architectures are AlexNet, GoogLeNet, VGGnet and ResNet [16]. The main insight from these models is that deeper models are better for classification. Based on these models for recognition, more models are proposed for other computer vision tasks. For example, R-CNN or Fast R-CNN have been proposed for object detection. Another widely used architecture is FCN for semantic segmentation. It is a fully convolutional neural network that can perform pixel wise labeling. The idea delves further into a deeper problem of structured prediction when recurrent neural networks can be seen as a generic sequence model like CRFs.

IV. Research Methodology

A. Research Philosophy

The philosophical stance of the research is highlighted in the research onion’s outermost layer. According to (Saunders et al., 2012) [16] there are four different philosophical branches that define the presence of a research entity; the first is positivism, the second is realism, the third is interpretivism and the fourth is pragmatism.

On the basis of empirical evidences and prior theories on brand management, brand choice frameworks will be examined in the current research. Positivism is mainly based on strong observation and forecasting outcomes similar to a laboratory scientist, with the aim of obtaining law-like generalizations for ascertaining cause and effect.

The researchers who adopt this approach underline the use of ‘scientific method’ for proposing and testing theories that have highly measurable and structured data, wherein the values of the researcher do not influence the research. Thus, this approach supports large samples of quantitative data which is analysed along with statistical testing of hypothesis. Such an approach helps test a theory, confirm a theory or revise a theory based on the analysis of the existing data.

This paper proposes positivism.

B. Research Design

The research design is experimental. The proposed navigation system employs a Google 3xL Smartphone. It will be used to continually capture images of the environment in front of a user and perform image processing and object identification to inform the user of the image results. The Uganda National Institute for Special Education (UNISE), Kyambogo University will be used as an experimental scene. UNISE is a national institute for students with special needs especially disabled students. On the other hand, the specification of Linux server hardware is a modern personal computer equipped with an i7 central processing unit (CPU) 64-bit i7 Intel/AMD-based PC and 4 gigabytes (GB) of RAM or higher and a graphics processing unit NVIDIA GeForce GTX 1050 GPU (or higher)to execute deep computing modules which are based on the faster region convolutional neural network (Faster R-CNN) algorithm.

C. *Software Design*

The software design of our proposed system involves the feature recognition, deep recognition, and direction and distance modules. There will be also a mobile application for interfacing the above modules. These modules and the application will be developed using Open CV, Java and Python computer programming languages.

D. *Experimental Data collections*

The experimental tests will be carried out on various candidates in the University. The tests will based on gender, age, degree of visual impairment, past experience on the use of similar or related equipment, literacy level, type of obstacle during movement and duration of time during the use of the experiment. Practically it is proposed that Women, children and the elderly may be given priority. Interviews may be carried out in assessing the performance, accuracy and speed of the equipment. The findings will be used to improve the performance and use of the equipment. About the experiment process, the participants will be required to turn the camera lens of smartphone to the front side and walk through institute campus from a main building through the parking towards the institute main gate. Each experiment will be arranged in the morning, noon and evening time. Before the experiment is carried out, a training session involving stakeholders can be arranged on how to use the equipment and precautions noted. This process would allow each participant to know the usage of the system. The selected participants may include five female and three male visually impaired students with a range of 17-25 years and also four old people like two females aged between 50-60 years and two males aged between 60-70 years. To obtain the degree of accuracy the degree of visual impairment should be similar. It should also be noted that this system will work for people with a hearing sense. The blind and deaf will not be managed by the proposed system. It should also be noted that the system will work for virgin disabled people who have never used any other system.

V. Research Product

The intended outcome of this research paper is a solution which solves the problem of movement for the blind people. This solution is a Smartphone-based guiding system for solving the navigation problems for visually impaired people and achieving obstacle avoidance to enable visually impaired people to travel smoothly from a beginning point to a destination with greater awareness of their surroundings. Blind people find it hard to walk through busy roads and travel new places and so the guided Smartphone will become their daily companion. This product is simple, cheap, user friendly and it is designed and implemented to improve the mobility of both blind and visually impaired people in a specific area.

VI. Conclusion

This paper proposed the development of a user-friendly guidance system for the visually impaired people. This system involves an advanced Smartphone and a Linux server connected and processed using a deep learning algorithm for image recognition. When the system is in use, the smart phone would continuously transmits images of the scene in front of the user to a server through using a 4G technology or a Wi-Fi network. Subsequently, the server performs the recognition process and the final results are transmitted back to the smart phone. The system would provide the user with obstacle track and avoidance information through voice notifications.

In the future, to provide information on more types of obstacles and more accurate recognition, a broader range of obstacle images and a high-end server equipped with a more powerful graphics processing unit could be used to increase the number of recognition categories and the recognition rate. The system is recommended for the blind people with a hearing sense and suitable for less traffic environments like universities, prisons and hospitals not busy city roads.

Acknowledgments

This research was partly supported by Kyambogo University Staff development committee under the Staff development vote. The lectures, guidance and comments from Prof Josephine Nabukenya of the department of Information systems, Makerere University, college of computing and informatics that helped the writer to furnish up the paper.

I wish also to acknowledge my fellow PhD students at Makerere students in the departments of computer science-Artificial intelligence lab, and department of computer networks for the various seminars conducted weekly on Thursdays which motivated me to carry out these research ideas.

Finally I wish to thank my reviewers who were appointed by the Federated Conference on Computer Science and Information Systems (FedCSIS) (DS-RAIT'19) Program Committee who reviewed this paper when it was submitted. I found your comments, directions and criticisms so important to furnish up this final revised position paper.

References

[1] A. S .Al-Fahoum., H. B .Al-Hmoud., and A. A.Al-Fraihat, "*A smart in-frared microcontroller-based blind guidance system*", Active and Passive, Electronic Components, vol. 2013

[2] V. Adagale and S. Mahajan,*" Route Guidance System for Blind People Using GPS"* and GSM. IJEETC ,4,16–21, 2015

[3] R. R. A. Bourne , S. R. Flaxman, and T. Braithwaite "*Magnitude, temporal trends, and projections of the global prevalence of blindness and distance and near vision impairment: a systematic review and meta-analysis".* Lancet Glob Health. 5: e888-e897, 2018

[4] S. Chaurasi and V.N. Kavitha, "*An Electronic Walking Stick for Blinds, in Information Communication and Embedded Systems*" (ICICES), 2014 International Conference on. IEEE, 2014, pp. 1–5

[5] V. Filipe, F. Fernandes, H. Fernandes, A. Sousa, H. Paredes, J. Barroso, *"Blind navigation support ystem based on Microsoft Kinect. In Conf. Proceedings of the 2012 International Conference on Software Development for Enhancing Accessibility and Fighting Info-Exclusion"* (DSAI), Douro, Portugal, pp. 94–101. 2012

[6] T. R. Fricke, N. Tahhan, E. Resnikoff,, A. Burnett, S. M. Ho, T. Naduvilath, K. S. Naidoo. "*Global Prevalence of Presbyopia and Vision Impairment from Uncorrected Presbyopia: Systematic Review, Meta-analysis, and Modelling. Ophthalmology"*. 2018 Oct; 125(10):1492-1499. doi: 10.1016/j.ophtha.2018.04.013

[7] V. N Hoang,T H. Nguyen, T. L. Le, T. H Tran,T. P Vuong, N. Vuillerme, "*Obstacle detection and warning system for visually impaired people based on electrode matrix and mobile Kinect"*. Vietnam J. Comput. Sci.,4, 71–83. 2017

[8] H. C. Huang, C. T. Hsieh, C. H Yeh, "*An Indoor Obstacle Detection System Using Depth Information and Region Growth. Sensors"* 2015, 15, 27116–27141.

[9] S. L Joseph., J. Xiao., X. Zhang., B. Chawda, K. Narang., N. Rajput., S. Mehta., L.V. Subramaniam, "*Being Aware of the World: Toward Using Social Media to Support the Blind with Navigation"*. IEEE Trans. Hum. Mach. Syst. 45, 399–405. 2015

[10] A. M Kassima., T. Yasunoa., M. S. M. Arasb., A. Z Shukorb, H. I. Jaafarb., M. F. Baharomb., F. A. Jafarb,. "*Vision Based of Tactile Paving Detection in Navigation System for Blind Person"*. J. Teknol. (Sci. Eng.) , 77, 25–32. 2015

[11] S. Mann, J. Huang, R. Janzen, R. Lo, R. Ramoersadm, V. Chen, A. Doha, *"Blind Navigation with a Wearable Range Camera and Vibrotactile Helmet"*. In Conf. Proceedings of the 19th ACM International Conference on Multimedia, Scottsdale, AZ, USA, pp. 1325–1328. 2011

[12] J. Malik, A. Arbeláez, C. Joao, F. Katerina., G. Ross, G. Georgia, S. Gupta, H. Bharath, A. Kar., and T. Shubhami. "*The three Rs of computer vision: Recognition, reconstruction and reorganization" Pattern Recogn*. Lett. 72 , 4–14. 2016

[13] A. Pereira., N. Nunesa., D. Vieiraa., N. Costaa., H. Fernandesc., J. Barroso, "*Blind Guide: An ultrasound sensor-based body area network for guiding blind people*." Procedia Comput. Sci., 67, 403–408. 2015

[14] M. Saunders., P. Lewis and A. Thornhill, "*Research Methods for Business Students"*. Pearson Education Ltd., Harlow. 2012

[15] R. Socher , C.D. Manning, and A.Y. Ng, " *Parsing natural scenes and natural language with recursive neural networks*." In Conf. 28th International Conference on Machine Learning (ICML-11). 129–136. 2011

[16] C. Szegedy, L. Wei , J. Yangqing, S. Pierre, R. Scott., A. Dragomir., E. Dumitru, V. Vincent., R. Andrew "*Going deeper with Convolutions*." in IEEE Conference on Computer Vision and Pattern Recognition, 1–9, 2015.

[17] B. Vauquois, "*A Survey of Formal Grammars and Algorithms for Recognition and Transformation in Mechanical Translation*." In Conf. Proceedings of IFIP Congress, 1114–1122. Edinburgh. 1968

[18] M. R Walter, M. Antone, E. Chuangsuwanich, A. Correa, R. Davi, L. Fletcher, E. Frazzoli, Y. Friedman, H. P. Jonathan, H. Jeong, S. Karaman, B. Luders, J. R. Glass, "*A situationally aware voice-commandable robotic forklift working alongside people in unstructured outdoor environments*." J.Field Robot. 32, 4, 590–628. 2015. DOI: 10.1002/rob.21539

[19] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, Kate Saenko, and T. Darrell. 2015. *Long-term recurrent convolutional networks for visual recognition and description.* In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2625–2634.

[20] H. Yu, N. Siddharth, A. Barbu, and J. M. Siskind. *A compositional framework for grounding language inference, generation, and acquisition in video*. J. Artif. Intell. Res. (2015), 601–713.

[21] H. Xu, S. Venugopalan, V. Ramanishka, M. Rohrbach, and K. Saenko. 20. *A multi-scale multiple instance video description network*. arXiv preprint arXiv:1505.05914 (2015).

[22] S. Venugopalan, H. Xu, J. D.Marcus Rohrbach, R. Mooney, and K. Saenko. *Translating videos to natural language using deep recurrent neural networks*. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015

Author Index