Infocommunications Journal

A PUBLICATION OF THE SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS (HTE)

December 2020 Volume XII Number 4 ISSN 2061-2079

MESSAGE FROM THE EDITOR-IN-CHIEF

PAPERS FROM OPEN CALL

CALL FOR PAPERS / PARTICIPATION

ADDITIONAL

Technically Co-Sponsored by

IEEE ComSoc™
IEEE Communications Society

hte

IEEE
HUNGARY SECTION

Editorial Board

Indexing information

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

Infocommunications Journal

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

Supporters

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

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

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

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

Publisher: PÉTER NAGY

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

www.infocommunications.hu

Booming usage of Infocommunications technologies brought to you by 2020

Pal Varga

WHILE the year 2020 was sorrowful in many ways, some of its challenges has been successfully overcome by humanity. Most of these successes have depended on Infocommunications technologies: these allowed us to keep the human interaction flowing during lock-downs. The most visible change was literally the broad usage of multimedia communications. Our communication spaces went ”virtual”: from schoolrooms through company meetings to massconferences; even the corridor- and coffee-break discussions got some alternatives.

On the other hand, we did not get easy access to laboratory or on-site physical equipment, so instead of building physical demonstrators and proofs, simulations and theoretical work must have received more focus even for those scholars who are used to hands-on experiences. Challenges are good for proceeding further, even if we didn’t choose to face such challenges. The ICT sector has received a new boost from the mass home-office working scenarios, virtual conferencing and in general, remote operations. These spins the concept of digital twins and cyber-physical systems further, and fuels various areas from satellite communications to cryptography, from new paradigms in function virtualization to information sharing frameworks in supply chains.

The following paragraphs provide brief overviews of the papers in the current issue of the Infocommunications Journal.

In his paper, our distinguished author, János Ladványszky presented efficient way of noise reduction in the form of a a modified Costas loop. The basic version of the Costas loop has been developed for demodulating SSB SC signals – the same circuit is applied for QAM demodulation as well, with decreased noise sensitivity. The two solutions applied in order to reach this is the transformation of the real channel input into complex signal, and the application of our folding algorithm. There are further improvements introduced in this paper, namely the complex channel input signal is normalized, and rotational average is applied. This results that at SNR of 0 dB, the solution provides 100 times better SER than that the original Costas loop does.

Categorizing the words according to their dictionary entry is a complex task even for contemporary text, whereas for less conventional texts (in old or less researched languages) it is even harder to solve this problem automatically. In their paper, Béla Benedek Szakács and Tamás Mészáros created an expert system-based automatic tagger that can be used to pre-process texts for dictionary-expansion. They demonstrated that a threecomponent tool performs better on Mikes Kelemen’s writings that are in an archaic dialect of the Hungarian language.

They compared their results to some already existing tools on the same corpus, and also created their own software that helps in expanding a dictionary containing word forms and tagging unprocessed text. Although this new tool was specifically designed for this task, it can be used in many other applications, and its flexibility allows for processing other noncontemporary or otherwise drastically different dialects.

In their paper, Dávid Haja, Zoltán Richárd Turányi and László Toka proposed a FaaS system design that offers horizontal scaling at the performance promise of host internal operation. Their proposed platform offers scaling to several server machines, when more compute power is needed. Besides, it minimizes the need for remote data access and remote function invocation through intelligent placement of data and function executions. They found that while not all application setups permit the reduction of remote operations under highload, partially because it is not possible to systematically reduce the fraction of remote operations, still, most of the computing problems exhibit clusters of often communicating functions and their data. One of their main poits is that these applications run faster in a FaaS platform considering locality.

Dániel Kozma and Pal Varga presented a newly designed development methodology for IoT-based Supply Chain Management (SCM) frameworks and platforms. They examine the main SCM modeling approaches and identify IoT-based frameworks and platforms to support Dynamic Supply Chains. They present an IoT-SCM development methodology, which defines these platforms’ characteristics by three main activities, namely the Collaboration, Control and Combination, and their related sub-activities. The methodology has been validated by the SOA-based IoT framework, Arrowhead.

Infocommunications Journal wishes you a Happy New Year with these articles – hope the number 21 will give us a lot more luck (and success) than 20 did.

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

Noise reduction for digital communications – A modified Costas loop

János Ladvánszky[1]

***Abstract*—An efficient way of noise reduction has been presented: A modified Costas loop called as Masterpiece. The basic version of the Costas loop has been developed for SSB SC demodulation, but the same circuit can be applied for QAM demodulation as well. Noise sensitivity of the basic version has been decreased. One trick is the transformation of the real channel input into complex signal, the other one is the application of our folding algorithm. The result is that the Masterpiece provides a 4QAM symbol error rate (SER) of $6^{*}10^{-4}$ for input signal to noise ratio (SNR) of -1 dB. In this paper, an improved version of the original Masterpiece is introduced. The complex channel input signal is normalized, and rotational average is applied. The 4QAM result is SER of $3^{*}10^{-4}$ for SNR of -1 dB. At SNR of 0 dB, the improved version produces 100 times better SER than that the original Costas loop does.**

***Index Terms*—noise, symbol error rate, QAM, Costas loop, Hilbert filter, folding algorithm.**

I. Introduction

Noise reduction is an important problem in communications. Digital communications are also sensitive to the noise. Effect of the noise can be detected by the symbol error rate (SER) as a function of signal to noise ratio (SNR). A possible circuit for noise reduction in digital communications is the Costas loop [1] whose original version has been developed for SSB SC demodulation. Essentially the same version can be used for 4QAM (Fig. 2).

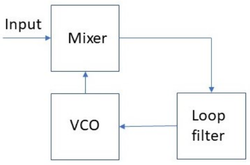


Fig. 1. Phase locked loop. The VCO output phase is related to the phase of the input signal. A simple modification can be used for frequency multiplication

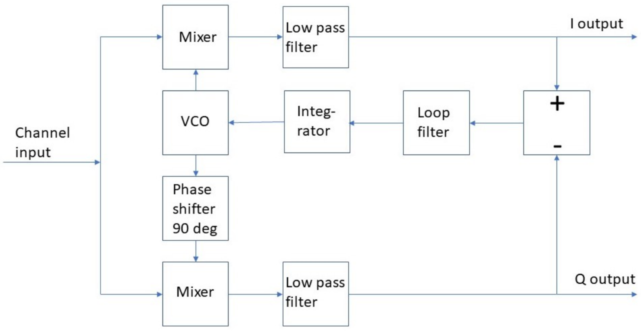


Fig. 2. Costas loop for 4QAM demodulation

Costas loop has been formulated from the phase locked loop (PLL, Fig. 1) [1] with introduction of separate branches for I and Q signals. A combination of the I and Q signals is used as VCO driving signal, and the two mixers have been supplied by the same VCO output signal and its phase shifted version, respectively. To understand the details of operation and its analytical treatment, please refer to [2].

The problem is that this Costas loop version is noise sensitive. Several tricks can be applied to decrease its noise sensitivity. Here we list them and apply some of them simultaneously.

Complex Costas loop

Real Costas loop is known primarily for SSB demodulation. Complex Costas loop is intended basically for QAM demodulation. From the real input signal, an analytical complex signal is formulated using Hilbert filter. Similarly, analytical version of the VCO signal is formulated. Accordingly, Complex Costas loop comprises a complex mixer and VCO signal also should be complex. In other respects, structure is the same as that for real Costas loop. Basic advantages are that BER can be better at the same value of SNR.

Averaging method

This is a method for stopping the rotation of the constellation diagram. In the VCO drive branch, signal is averaged in parallel using two different time constants. If the results are the same, then the constellation diagram stops rotation.

4th power method

Used for carrier recovery of 4 QAM. If the receiver input signal is raised to the 4th power, then the four constellation points are transformed into the same point. That means, in one step, all information has been removed but the carrier. Advantage is very exact reproduction of the carrier. Noise sensitive.

Pulse counting method

For stopping rotation of the constellation diagram. Horizontal and vertical projections of the rotating constellation diagram contain extra steps compared to the case without rotation. Making pulses from steps by differentiation and counting and minimizing the number of steps, can be used for stopping rotation.

Folding method

Very much noise insensitive. Replaces 4th power method. Constellation diagram is folded along an axis then the result is shifted into a symmetric position with respect to the origin. This step is repeated until one point (the carrier) remains. This method can be used for real Costas loop as well, and for

[1] Formerly with Ericsson Hungary, Budapest, Hungary. (e-mail: Ladvanszky55@t-online.hu)

DOI: 10.36244/ICJ.2020.4.1

QAM of arbitrary degree. BER of 0.01 is possible at SNR of -4 dB.

Normalization

Used before correlation. Complex signal is normalized exploiting that exp(jωt) has an absolute value of 1. Cannot be used for real signal.

Limitation

of the VCO drive signal. Used for stopping rotation, especially in large noise. We observed that adding a large noise to the useful signal at the input of the Costas loop, significantly increases VCO drive signal thus causing rotation. Limitation of the VCO signal from below and above, limits the effect of the noise on the VCO signal.

QAM sc

It is observed that carrier in the receiver input signal interferes with the carrier produced by the Costas loop. Thus carrier (and possibly one sideband) at the receiver input has been removed by a filter.

Correlation method

Used for stopping rotation. QAM signal is produced in two different ways and the results are correlated. Deviation of the correlation coefficient from 1 is used as VCO drive signal.

Differential coding

Used for stopping rotation. Differential coding is not affected by rotation. We code the modulation signal with differential coding, and after demodulation, we use the same code for decoding [5].

II. APPLICATION OF COMPLEX INPUT SIGNALS

Basic version of the Costas loop is changed by inserting a block between the channel and the input of the Costas loop [2] (Fig. 3). Essence of the change is application of complex signals [2]. However, in [2], the advantages are not fully exploited. We add normalization of the input signal, that has a significant effect on noise reduction.

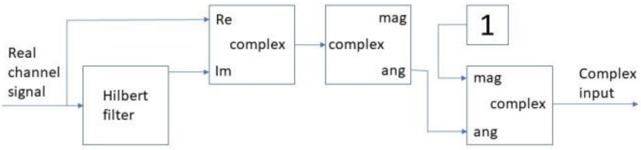


Fig. 3. Transformation of the real channel signal into a normalized complex signal

It is widely known that in order to produce an analytic signal, imaginary part of the signal can be formulated by application of a Hilbert filter for the real signal [1]. Narrow-band approximation of a Hilbert filter is a 90 deg phase shifter or the corresponding delay circuit.

To remove a part of the noise from the complex signal, it is normalized by setting its absolute value to unity. Effect of application of a complex signal and its normalization has been shown in Fig. 4.

Because of insertion of the block into the Costas loop, a complex mixer must be used instead of the two real mixers, the VCO signal must also be complex and there is a modification at the beginning of the branches. We detail these modifications in Section V.

III. THE FOLDING ALGORITHM

Folding algorithm [3] means two foldings for 4QAM, one across the real axis and another one across the imaginary axis (Fig. 5-7). As the noise is different around all points of the constellation diagram, folding algorithm averages noise. Folding algorithm is applicable for higher order constellation diagrams as well. We consider here 4QAM only.

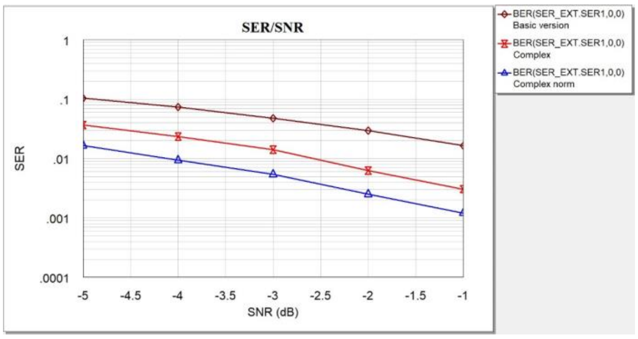


Fig. 4. Effect of application of a complex signal and its normalization on noise properties. Three curves for SER vs. SNR are shown. The upper curve is without complex signal. The middle curve is with complex signal but without normalization. Bottom curve is with normalization

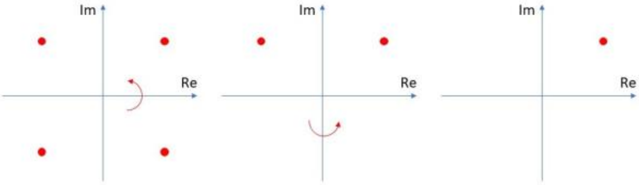


Fig. 5. Explanation of the folding algorithm for 4QAM. Left: The original 4QAM. Middle: After a folding across the Re axis. Right: After a folding across the Im axis. Only one point remains, it is perfect for carrier recovery

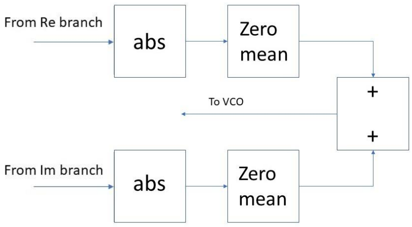


Fig. 6. Part of the system realizing the folding algorithm

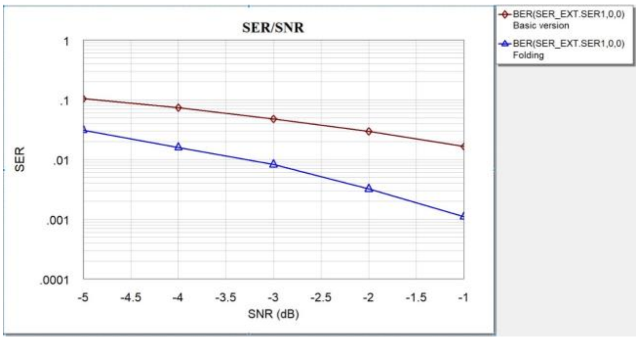


Fig. 7. Result of the application of the folding algorithm. Upper curve: Basic Costas loop, lower curve: With folding algorithm

IV. APPLICATION OF THE ROTATIONAL AVERAGE

Based on the right graph in Fig. 5, a new idea occurs. The noise can also be averaged after folding algorithm, if the noise in the neighborhood of the remaining constellation point is rotated around the point. We try one 90 deg rotation as shown in Fig. 8-10, but the number of rotations can be arbitrary.

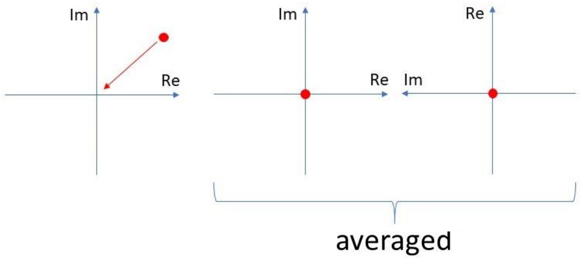


Fig. 8. Sketch of application of rotation

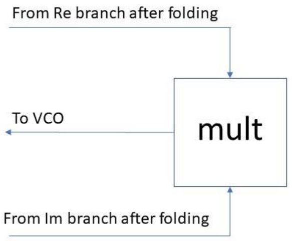


Fig. 9. Part of the system realizing rotational average. Explanation: $\left(\frac{Re+Im}{2}\right)^2-\left(\frac{Re-Im}{2}\right)^2=Re*Im$. Not jIm, this is not an error.

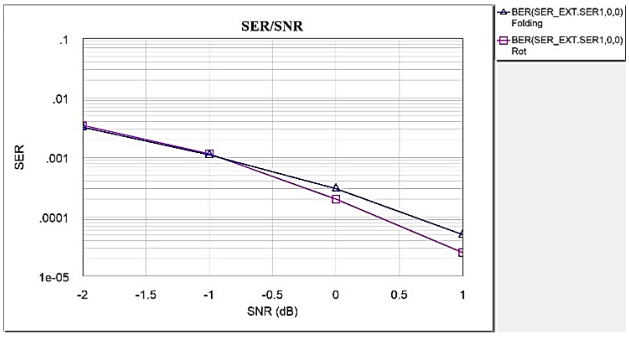


Fig. 10. Result of application of rotational average. Upper curve: Folding algorithm alone, lower curve: With application of rotational average. For bad SNR, the two algorithms offer approximately the same performance. But at slightly better SNR, the advantage of the rotational average is obvious

V. THE IMPROVED MASTERPIECE

First, we show the schematics including complex signals with normalization, folding, and rotational average (Fig. 11, at the end of the paper). Noise properties are shown in Fig. 12.

VI. CONCLUSIONS

In this paper an effective method for noise reduction for 4QAM communications has been shown. Other Hungarian efforts on decreasing the effect of interference and noise are found in [6]. Our intention is application of this circuit in our version of quantum communication system.

Main statements are the new modifications of the Costas loop for achieving better noise properties: Using complex, normalized input signal (Fig. 3), the folding method (Fig. 6) and the rotational averaging (Fig. 9). These methods may result in breaking the Shannon formula: The most recent version can also work at SNR=-22 dB [7].

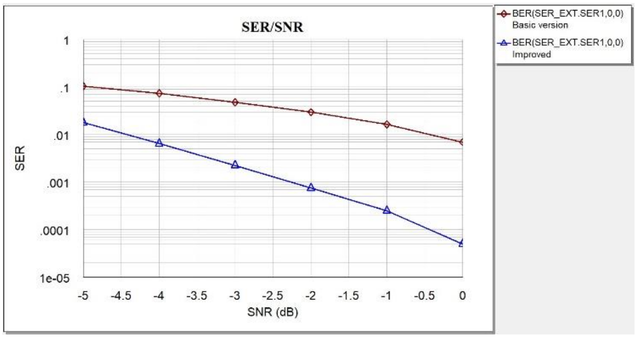


Fig. 12. Noise properties of the improved Masterpiece compared to the basic Costas loop

VII. ACKNOWLEDGMENTS

This research was started with a discussion with Dr. András Radványi at our sailing boat get together in Balaton lake in August 2017. He put especially interesting questions, many thanks for them.

Sincere thanks are due to Dr. Benedek Kovács, last colleague of the author at Ericsson Hungary before retirement, who made it possible for the author to work at home and he participated in this work as well as co-author of the patent [4].

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

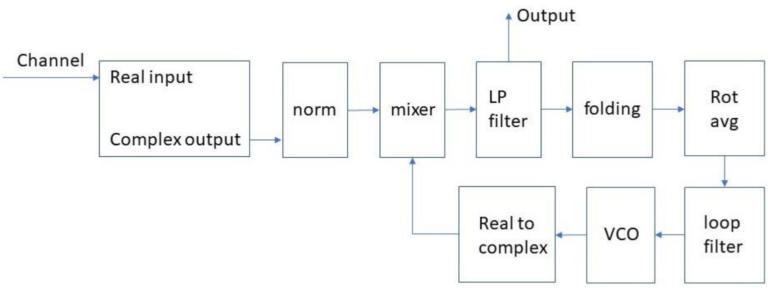


Fig. 11. Block diagram of the improved Masterpiece, a modified Costas loop

VIII. References

[1] J. G. Proakis: “Digital Communications”, McGraw-Hill, 2001

[2] R. E. Best, N. V. Kuznetsov, G. A. Leonov, M. V. Yuldashev, R. V. Yuldashev: „Tutorial on dynamic analysis of the Costas loop”, Annual Reviews in Control 42 (2016) 27–49

[3] J. Ladvánszky: “A Costas loop variant for large noise”, Journal of Asian Scientific Research, Vol. 8, No. 3, 144-151

[4] J. Ladvánszky, B. Kovács: “Methods and apparatus for signal demodulation”, Ericsson patent, 2018.03.01. P74032 WO1; PCT application date: 3/1/2018; application Nr. PCT/SE2018/050198

[5] J. Ladvánszky: ”A Costas loop with Differential Coding”, International Journal of Contemporary Research and Review, ISSN 0976 – 4852, October, 2017|Volume 08|Issue 10

[6] A. Hilt, G. Maury, B. Cabon: “Radio-frequency interference in digital communication links”, Híradástechnika/Journal on C5, 11/1999

[7] J. Ladvánszky: “A modification to the Shannon formula”, Network and Communication Technologies; Vol. 5, No. 2; 2020, DOI: 10.5539/nct.v5n2p1

János Ladvánszky (Senior Member, IEEE) was an electrical engineer at Ericsson Hungary (MSc in 1978, „Modelling of a microwave transistor”, from the Budapest University of Technology, PhD in 1988, „Nonlinear, microwave circuit design”, from the Hungarian Academy of Sciences, DSc defense is just coming, „Integrated systems for optical communications”, at the Hungarian Academy of Sciences). He is the world champion in 2018 in the competition on writing scientific papers in electrical engineering. Recently he got retired. His biography appeared in the 2019 edition of the book ”Successful persons in Hungary”, by Britishpedia. His carrier can be followed at the ResearchGate, the LinkedIn and the Facebook. At the time of the submission of this paper, he has 6 books, more than 180 publications, 14 patents, 135 citations, and about 8300 reads at ResearchGate.

Hybrid Distance-based, CNN and Bi-LSTM System for Dictionary Expansion

Béla Benedek Szakács and Tamás Mészáros

Abstract— **Dictionaries like Wordnet can help in a variety of Natural Language Processing applications by providing additional morphological data. They can be used in Digital Humanities research, building knowledge graphs and other applications. Creating dictionaries from large corpora of texts written in a natural language is a task that has not been a primary focus of research, as other tasks have dominated the field (such as chat-bots), but it can be a very useful tool in analysing texts. Even in the case of contemporary texts, categorizing the words according to their dictionary entry is a complex task, and for less conventional texts (in old or less researched languages) it is even harder to solve this problem automatically. Our task was to create a software that helps in expanding a dictionary containing word forms and tagging unprocessed text. We used a manually created corpus for training and testing the model. We created a combination of Bidirectional Long-Short Term Memory networks, convolutional networks and a distance-based solution that outperformed other existing solutions. While manual post-processing for the tagged text is still needed, it significantly reduces the amount of it.**

Index Terms—**machine learning, convolutional neural network, bidirectional LSTM, Levenshtein-distance, dictionary.**

I. Introduction

THE task of creating a dictionary from a corpus is a complex one that requires a lot of manual labour without a sufficiently accurate automatic tool, and even with that some amount of manual post-processing is still needed, as most solutions do not provide 100% accuracy.

Automatic dictionary expansion has been a task used in various fields [1] such as biomedical data [2]. Sometimes a human-in-the-loop approach is applied [3], somewhat similar to what our research led to.

One-language dictionaries such as Wordnet [4] have been used extensively in NLP (Natural Language Processing) research. They can provide information about the text's vocabulary, can serve as a basis for knowledge graphs and other applications.

It is important that this task is analogous to stemming or lemmatizing, but that only covers part of the problem: there are headwords that have multiple meanings in a language, and a proper dictionary expansion tool should be able to decide between them in addition of finding the correct headword form.

These problems increase when we are dealing with non-conventional texts: either in languages that does not have a wide array of tools and research in terms of NLP or texts in significantly different dialects (old texts or texts of highly specific environments, such the language of online communities). In these cases, previously used algorithms and tools will provide results that will be too inaccurate for any applications.

If there is a sufficiently large corpus of text from the specific dialect we are focusing on *and* it is processed by hand, it can be enough to teach some kind of model on it. This was our approach in this case: we were trying to develop a software specifically tailored to expand an already existing dictionary in a specific format. In this case, we had two main tasks to solve: looking at a word, we had to find the corresponding form in the dictionary, or the headword if the form does not exist, and if there are multiple forms, decide which one is the most likely based on the context.

II. The Mikes Dictionary Project

This is a project [5] that was created by the Hungarian Research Center for the Humanities, and the main purpose is to create a full author's dictionary [6] based on the work of Kelemen Mikes, an 18th century Hungarian writer, who had a large body of work comprised of mostly prose and letters. The researchers will be using this dictionary for a multitude of analytical experiments in the field of Digital Humanities [7].

A. The Corpus

Kelemen Mikes was a very influential writer in the 18th century, and his work is still extensively studied. The language of Mikes is, however, very different from contemporary Hungarian: the grammar is much more inconsistent, he uses a lot of Latin words and expressions. The dialect which he uses is mostly understandable by a contemporary reader, but only because of the flexibility of the human mind.

This also means that most tools developed for processing contemporary Hungarian are significantly less accurate on these texts, so we had to create a different solution.

This paper was submitted on 2020.09.29.

Béla Benedek Szakács is with the Budapest University of Technology and Economics, Budapest, Hungary (e-mail: benedek.b.szakacs@gmail.com).

Tamás Mészáros is with the Budapest University of Technology and Economics, Department of Measurement and Information Systems, Budapest, Hungary (e-mail: meszaros@mit.bme.hu).

DOI: 10.36244/ICJ.2020.4.2

B. *Manual Work*

We have been involved with other experiments concerning Mikes's texts before, and it has been fully digitalized with the proper notations. The dictionary project, however, is only partially done.

The current state of the dictionary was created manually by linguists at Hungarian Research Center for the Humanities. This was an enormous task, even for one part of the whole Mikes corpus, called the "Turkish Letters", a collection of 207 letters (~106 000 words). They are only a fraction of Mikes's complete work, but a large enough body of text to use as a solid basis for training algorithms.

Because of the heavily time-consuming nature of the manual work, our task was to significantly increase its speed by developing an automatic tagger software that would predict the headwords for the words, and afterwards a researcher would correct the mistakes manually. With even a moderately high accuracy this would significantly increase the speed of work on the corpus (we are talking about years of manual labour). This human-in-the-loop approach can be compared to other machine-assisted manual works, such as Alba et. al. 2019 [8] or Ruis et. al. 2020 [9].

C. *Automatic Dictionary Expansion*

The manual work on the dictionary, even with the help of a simple software that allows fast tagging of words with dictionary entries and offers help based on purely by the existing dictionary, is a very time-consuming task, requiring significant expertise in linguistics. This means that to meaningfully increase effectiveness, the software should contain an automatic tagging tool.

The goal is to process the entire unprocessed corpus, and one of the most important inspirations for developing this software was to help this work, creating some kind of pre-processing application that allows linguists to quickly decide whether the tagging created by the software are correct or not, and make it accurate enough that it provides sufficient help. We focused on developing the algorithmic part of this task, the problem of automatically tagging words with predicted dictionary entries.

This is a very difficult task mostly because of the language used: even the linguists doing the manual work have difficulties quickly discerning the headword because of the archaic forms and the foreign (mostly Latin) words. This means the task of doing it automatically was expected to be a highly complex task. We should not expect as high accuracy scores as with state-of-the-art tools on contemporary texts. The goal was rather to achieve a sufficient score that provides significant help as pre-processing.

It is important to mention that there are words with multiple parts that can be separated in the text but should be recognized as one word. The recognition and categorization of these words is a very difficult task that this solution is not able to solve, so words with multiple parts are not recognized as one word, diminishing the accuracy of the tool. This is a known shortcoming of this solution.

There are also headwords that are identical in form but carry different meanings. Since in our pre-existing corpus, the example sentences are separate for each headword, we treated these similar headwords as completely different entities. This also means that tools not using the context of the word will not be able to identify it correctly.

III. Existing Solutions

While we expected other tools developed mainly for analysing contemporary texts to underperform, we have nevertheless inspected an array of other tools commonly used in Hungarian NLP tasks.

We were not trying to take all tools into consideration, only a select few. We were trying to use software from widely different backgrounds: state-of-the-art solutions as well as old but reliable ones.

We performed experiments with four solutions that can be divided into three groups: state-based, reliable solutions, out-of-the-box, performance-focused tools, and state-of-the-art models.

With this, the four solutions we have chosen are the EmMorph morphological analyser [10], the SpaCy NLP pipeline, the BERT [11] (Bidirectional Encoder Representations for Transformers) and Flair [12] models, using the Flair framework.

A. *EmMorph*

This tool was created by Attila Novák [10], and it is a finite state machine-based tool trained on contemporary Hungarian. Because of this, it is an extremely accurate and reliable tool, as it is not a probability-based model.

It uses an "Item and Arrangement" (IA)-style analysis, so the input word is analysed as a sequence of morphs, where each form is a specific realization (an allomorph) of a morpheme. This means that the EmMorph does a very detailed analysis of each word, providing a lot of morphological information.

Because this solution does not use any form of probability, it relies on its database for every word analysed. This means that new word forms will be unrecognizable to it, rendering it a lot less effective in our case: the text from Mikes contains a lot of word forms not used in contemporary Hungarian.

B. *SpaCy*

SpaCy[1] is an industrial-strength, out-of-the-box solution in Python for NLP problems. It is designed for production environments, not for experimentation, although it is open source, and has a fair number of contributors, as well as a very flexible architecture that allows easy integration for custom components.

The underlying technology is mostly built on convolutional neural networks, but it uses embeddings and other pre-computing strategies. They do not have a definitive paper that summarizes their methods, instead they have a summary of the technology on their webpage.

[1] https://spacy.io/

SpaCy provides a variety of functions for NLP tasks: it has a lemmatizer, a PoS (Parts of Speech) tagger, a dependency parser that builds dependency trees between tokens in a sentence, an entity recognizer for NER (Named Entity Recongition), a built-in categorizer for text classification tasks, a pattern matcher, and can incorporate custom components. This allows for a variety of configurations based on the task at hand.

SpaCy also has a dictionary-like system that stores lexemes and data about the document's vocabulary, and it is capable of full morphological analysis, including noun cases, verb tenses and others.

C. BERT

BERT [11] is an acronym for Bidirectional Encoder Representation for Transformers. It was developed by Google, mostly for NLP tasks. It is basically a multi-layer bidirectional Transformer, trained on a very large corpus, resulting in a network that can be easily adjusted to any NLP task using just an extra layer and some fine-tuning. It relies heavily on the concept of transfer-learning, the concept of using a pre-trained model with little training on a specific dataset for a specific task. It is mostly utilized in tasks where training data is scarce or absent. For NLP, this means that BERT was trained extensively on a huge multilingual corpus unsupervised, and so it learns a lot of the characteristics of the language, making fine-tuning a lot faster and less data-extensive.

The main improvement from the precursor model is that they use bidirectional unsupervised learning. This allows it to be successful at a large variety of uses, including both token level and sentence level tasks. This bidirectional training relies on a method called MLM (masked LM) as to not run into the problem of the words "seeing themselves" (the word that needs to be predicted is present for the opposite direction, making the task trivial), by randomly masking words in both directions and trying to predict them.

D. Flair

The name Flair [12] is used for multiple things: it is both an NLP library (including a data library and pre-trained models for a variety of tasks), built on PyTorch, and an embedding model.

The Flair framework is designed to make using big, complex models very simple. It is a wrapper over PyTorch, one of the most widely used machine learning libraries for Python, and it makes creating for example, a BERT model for text classification extremely simple. It also has a variety of pre-trained networks for the most common tasks, such PoS tagging.

Flair itself is a character-level recurrent network using contextual string embedding, usually fed into a Bi-LSTM-CRF (Bidirectional Long-Short Term Memory, Conditional Random Fields) model. It is currently the best solution for PoS tasks, as it outperforms every other approach, including the previously mentioned BERT. However, it was specifically designed for sequence tagging, not for more complex tasks (although it can be used in other models designed for different tasks, as it is only an embedding).

IV. The Models

Both of our main tasks (finding the correct headword for unknown word forms and discerning the correct headword for unambiguous word forms) are essentially categorization tasks. We have focused on the first one, as doing it correctly technically includes the second one as well. This means that the output of the system should be one of the existing dictionary entries, whereas the input should be the word and some of its context.

Because of the strict form of example sentences (31 words, the middle one is the target for tagging), we decided for a fixed-length input, not a sentence-based one. This still allows for the procession of sentences, with the use of padding tokens, and the length of the input means that most sentences will be inside its bounds. While this special format made the system theoretically suboptimal, back-conversion of the dataset was practically impossible, but this format still allowed for keeping most of the word's context.

For the models themselves we focused on two features of the input: the words in the context (31 words) and the middle word's characters (maximum 40, 44 different possible characters). Because in the Hungarian language most words are similar to their headwords, and a character-level model is a great solution in looking for it.

The architectures we have chosen were the one-dimensional convolutional neural networks [13] and the Bi-LSTM [14]. Both of these have been extensively used in NLP tasks. In our setup, we have used two models, trained and evaluated separately: one using both character-level and word-level convolutional networks, and a CNN-Bi-LSTM solution using convolutional network for the character-level input but Bi-LSTM for the word-level input.

We have also experimented with a pure Bi-LSTM solution, but it underperformed compared to the CNN-Bi-LSTM solution and was deemed too similar to the CNN-Bi-LSTM solution to be used alongside it (more information about it can be found in the Training and Evaluation chapter).

A. Embeddings

In both cases, the input words are embedded in a simple 256-dimensional embedding, and the characters are one-hot encoded.

For most NLP applications, pre-trained embeddings are usually a staple. The problems with this approach in this case were that the unique language of Mikes' writing made it impossible to utilize any pre-trained embedding. We could have used embeddings taught on the data itself, but the size of the corpus was not sufficient for this task.

B. Pure Convolutional Model

The model is a straightforward convolutional model, with only one layer of 1D convolution (Fig. 1). Because the size of the training set was not enough for large, complicated language models, we opted for a smaller, simple model.

For optimizing the hyperparameters, we assumed that the parameters themselves can be independently optimized. This approach was necessary due to the large number of possible combinations. We selected dropout (from 0.0 to 0.5 with 0.1 increments, dense, convolutional and bi-LSTM layers optimized independently), batch size (values: 64, 128, 256, 512), the type of optimizer (Adam and Nadam), the type of

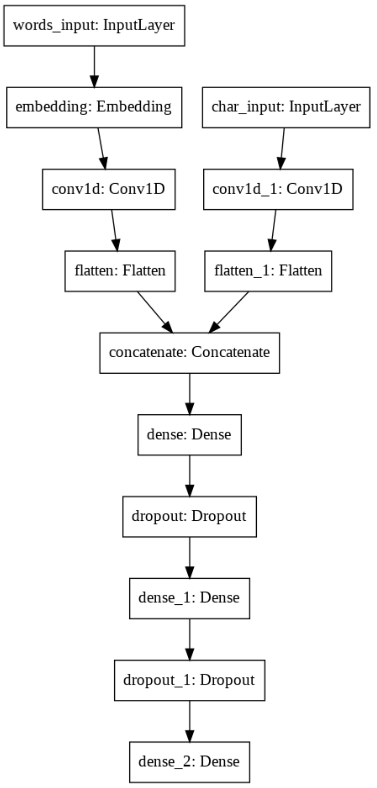


Fig. 1. The layers of the pure convolutional model

kernel initializer for all layers (values: uniform, normal, glorot_uniform, glorot_normal, lecun_uniform) and the size of the embedding layer (values: 64, 128, 256, 512) as optimizable hyperparameters. The best regularizer was Nadam, and the best batch size was 64 for both this and the CNN-Bi-LSTM model.

The optimised hyperparameters of the layers are the following (all other hyperparameters can be found in the Appendix chapter):

1. Embedding layer: input dimension = 31, output dimension = 256, embeddings initializer = ”lecun_uniform”, no regularization, no zero masking, no dropout.
2. Conv1D layers: filters = 29 for conv1d and 30 for conv1d_1 and conv1d_2 in the CNN-Bi-LSTM model, kernel size = 3, activation = relu, no dropout.
3. Dense layers: units = 512, 4096, 15829, activation = “relu”, dropout = 0.3.

The two Conv1D layers are concatenated in a 50-dimensional layer, and then a series of dense layers are responsible for increasing the dimension to the size needed for the output. Because the task is simple categorization, we used a simple softmax function at the end and sparse categorical crossentropy as the loss function. The dimension (15829) of the last layer is equal to the dictionary entries. While this means that subsequent additions to the dictionary means using a completely new model every time, the simplicity of the model means a low number of parameters (88 million), and with that a relatively fast training compared to the current, much larger models (more on that in the Training and Evaluation chapter).

Using convolutional layers in text processing is a commonly used technique ([15], [16]), although mostly used for sentiment analysis or text classification.

C. *CNN-Bi-LSTM Model*

The CNN-Bi-LSTM model (depicted on Fig. 2, with an example input and output) was inspired by another architecture primarily developed for Named Entity Recognition [17], although it lacks the case-embedding and uses a simple 256-dimensional embedding. It is very similar to the previously described pure convolutional model, the only difference is in the processing of the embedded words.

We used a 64-dimensional Bi-LSTM layer that was then flattened into a dense 64-dimensional layer. This meant that unlike with the pure convolutional model, here the output of the word-processing part of the model was a lot larger. We theorized that this, together with the LSTM being generally more fitted for processing word sequences, will lead to a better accuracy than the pure convolutional model.

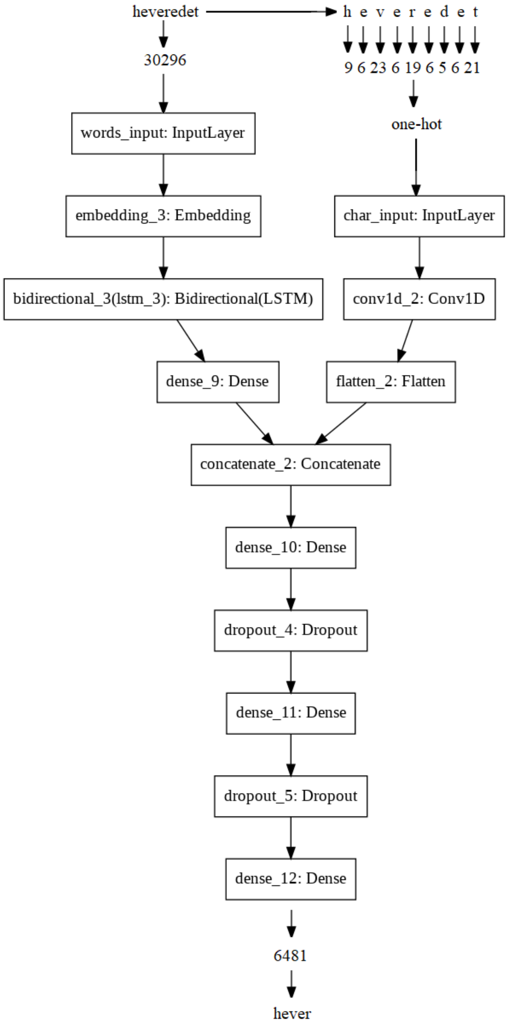


Fig. 2. The layers of the CNN-Bi-LSTM model and an example input and output

The hyperparameters of the Bi-LSTM layer were the following (the hyperparameters of the other layers are the same as the ones in the pure convolutional model, and the other hyperparameters can be found in the Appendix chapter):

Units = 64, activation = "tanh", recurrent activation = sigmoid, use bias = True, kernel initializer = "lecun_uniform", dropout = 0.1.

We expected the CNN-Bi-LSTM network to outperform the pure convolutional model, but also to learn more slowly. Also because of the difference in the word processing architecture, we theorized that the two networks would be better at identifying different words. This led to the final solution combining the result of both models and deciding between them. The problem was that we needed a three-opinion system to use majority voting, so we used a third component, a simple distance-based solution.

D. Levenshtein Distance

The Levenshtein distance is often used in approximate string matching, especially in spell checking, where one of the strings comes from a dictionary. This is somewhat similar to the task of finding a headword, although not equal. We have chosen this as an often used and simple solution in string distance measurement.

Levenshtein distance is a measurement of difference between two strings, based on the number of *edits* that are needed to transform one to another. These edits are: 1. adding a character, 2. deleting a character, 3. substituting a character with another character. This can be calculated very efficiently using a dynamic programming algorithm.

The main issue with using a distance-measurement like this in a dictionary of roughly 16 000 word is that every time we need to do the whole calculation 16 000 times. This, even with a C implementation, takes significantly longer than simply running one word and its context through the models for prediction. So, as we can see, using Levenshtein every time leads to a significant amount of time increase, which, while not necessarily one of the main considerations, is still a factor to keep in mind.

E. The Hybrid System

We have decided to use two models and the Levenshtein-distance as a three-part expert system. Because of the higher computational cost and the distribution of correct guesses (Fig. 3), the Levenshtein-distance was used as a tiebreaker.

The execution was very simple: first, both models predicted a dictionary entry, then if these were not the same, the entry predicted by the Levenshtein-distance was used, regardless of the results produced by the models.

While as we will see on Fig. 4, both the CNN-Bi-LSTM model and the convolutional model outperformed the distance-based prediction, a disparity in the results of the two models usually means they are both wrong, and in this case the distance can be a helpful third option. This is the reason why the Levenshtein distance takes priority over both of them.

It is not trivial that these three solutions complement each other well, but in this application the results show that the system together performs significantly better than the individual components, which can be explained by the varying architectures.

V. Training and Evaluation

To allow for a good evaluation, we have used the following method: we randomly chosen 5% of all known word forms as the test set and excluded them from the dataset on which we performed the training. This meant that the word forms used in testing are analogous to unknown words which the application has to predict headwords for. This set formed our testing set during the evaluation phase.

The training was always performed with 48 iterations, in every iteration a new set of 10 000 sentences were used to train the model with 10% as the validation set. We used early stopping based on validation loss with a patience of 3 and used the best weights.

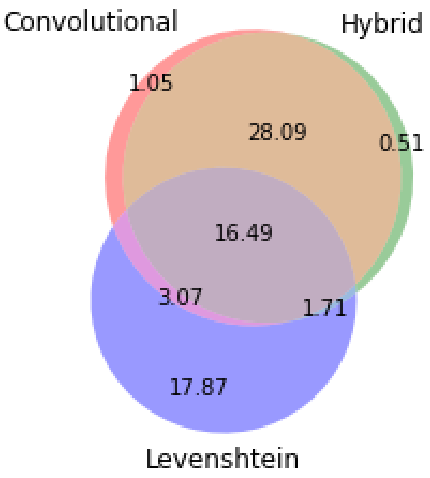


Fig. 3. Accuracy of the combination of components in percentage of accurately guessed headwords using the whole test dataset.

A. Evaluation

Pure Bi-LSTM stands for the previously mentioned model where even the character-level input was fed into a Bi-LSTM network. It did not achieve similar accuracy to the other two models, so it was discarded.

Contrast these results (Fig. 4) with the performance of the Levenshtein-distance-based solution: that achieved 37% on the same dataset. We used it as our baseline, as it is the most basic solution, and it performed remarkably well for its simplicity.

Type	Best Accuracy
CNN-Bi-LSTM	46.8%
Pure Bi-LSTM	37.5%
Pure convolution	48.7%
Levenshtein	37.0%
Hybrid System	65.9%
SpaCy	41.3%
EmMorph	21.9%
BERT/HuBERT	-
Flair	29.7%

Fig. 4. The accuracies of all solutions. BERT was not tested because training was early stopped at 19% accuracy. HuBERT achieved 20% accuracy on the training set.

Contrary to our expectations, the pure convolutional model had slightly better accuracy than the CNN-Bi-LSTM model. This can be explained with the length of the context fed into the input: only 10 other words were used as context, and Bi-LSTM networks are mostly used because of their ability to identify long-term connections. Furthermore, experiments have shown [18] that in certain sequence-labelling problems, CNNs can outperform RNNs. It is also worth mentioning that the gap in accuracy is very small (only 1.9%). Using both was, however, crucial for the system to be able to use majority voting.

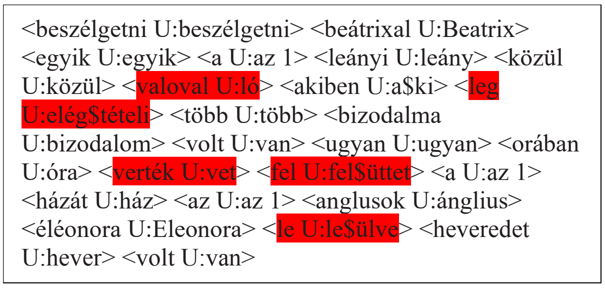
The system as a whole achieved 65.9% accuracy all together on the same dataset. This means that the parts of the system do, in fact, perform significantly better together (even the convolutional model which performed the best of all the solutions achieved only 48.7%).

We also experimented with different ways of deciding ties between the components, and found out that the Levenshtein distance was, in fact, the best for this task, despite being the least accurate standalone. Fig. 3 shows that the distribution of correct guesses supports this.

Training times were typically around 3-4 hours for the whole system. All training and evaluation were performed on a PC with a GTX 1060 6GB GPU. With a TPU using more VRAM training these models would take even less time.

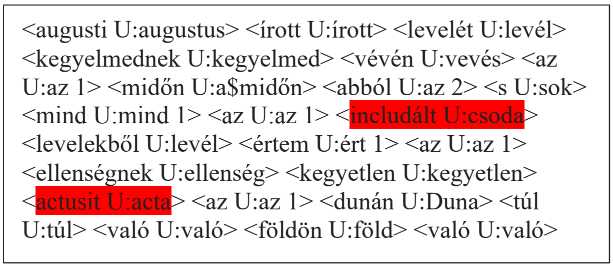
B. Testing on Other Texts

Whereas the previously mentioned experiments provide a numerical metric, in the case of a tool designed to help manual work, manual experiments were also needed. We have used it on a handful of other texts to manually test its usefulness and if any typical errors are present.



<beszélgetni U:beszélgetni> <beátrixal U:Beatrix> <egyik U:egyik> <a U:az 1> <leányi U:leány> <közül U:közül> <valoval U:ló> <akiben U:a$ki> <leg U:elég$tételi> <több U:több> <bizodalma U:bizodalom> <volt U:van> <ugyan U:ugyan> <orában U:óra> <verték U:vet> <fel U:fel$üttet> <a U:az 1> <házát U:ház> <az U:az 1> <anglusok U:ánglius> <éléonora U:Eleonora> <le U:le$ülve> <heveredet U:hever> <volt U:van>

Fig. 5. The results of testing on a letter from Mikes not from the “Turkish Letters”. The red background signals the wrong predictions. The $ symbol is the separator for multi-part words (e.g. “elég$tételi”), and for multi-meaning words, a number is appended (e. g. “az 1”)



<augusti U:augustus> <írott U:írott> <levelét U:levél> <kegyelmednek U:kegyelmed> <vévén U:vevés> <az U:az 1> <midőn U:a$midőn> <abból U:az 2> <s U:sok> <mind U:mind 1> <az U:az 1> <includált U:csoda> <levelekből U:levél> <értem U:ért 1> <az U:az 1> <ellenségnek U:ellenség> <kegyetlen U:kegyetlen> <actusit U:acta> <az U:az 1> <dunán U:Duna> <túl U:túl> <való U:való> <földön U:föld> <való U:való>

Fig. 6. The results of testing on a letter from Ferenc Rákóczi the II, a contemporary of Mikes. The red background signals the wrong predictions. The $ symbol is the separator for multi-part words (e.g. “a$midőn”), and for multi-meaning words, a number is appended (e. g. “mind 1”)

For experimenting on these texts, we used the hybrid system, including a lookup function that uses the Solr database to look for already existing forms (the Solr query searched for an exact match amongst the word forms, and we grouped them by headword, then in the case of multiple matching headword, used the prediction function). This means that the prediction function is only used in the case of ambiguous or unknown forms.

As we can see on Fig. 5. and Fig. 6., the number of wrong predictions is low, much lower than the measured 34%. This is primarily because these texts contain a lot of words that are already known. The large size of the dictionary allows for more common words with a small number of forms to be easily identified without the uncertainty present in the predictive function. The errors do not show any certain trend (some are similar to the wrongly predicted headword, some are seemingly random, some are the result of the model’s inability to correctly identify multi-part words).

C. Comparison to Other Solutions

We have performed the same experiment with the dataset used for training and testing on all the previously mentioned other solutions. For both the EmMorph and SpaCy we only used pre-trained models, and for the BERT and Flair we trained multilingual models on the training dataset.

EmMorph performed according to our expectations: with the score of 21.9%, it was the least accurate. It is mostly due to the large number of unknown words and a very different grammar used in Mikes’s writing.

SpaCy (using the lemmatizer module, and the lemma of each word as the predicted headword) has surpassed our expectations with its score of 41.3%, as it performed closely to our individual networks. This shows the power of context-analysing (EmMorph only uses the form of the word, SpaCy uses the context as well), and the robustness of its Hungarian models.

We have used a multilingual BERT model as well as the HuBERT [19] model. We used the Flair framework for training and evaluating the BERT implementations, using Flair’s own built-in categorizer architecture. We used ADAM optimizer with a learning rate of 0.1 and an annealing factor of 0.5, minibatch size was 32. The multilingual model’s raining was stopped at 19% accuracy, and the training of HuBERT was stopped at 20% accuracy. We have not evaluated them on the test dataset.

We used the Flair embeddings similarly to the BERT embeddings, in the same setup. It has performed much better than BERT, reaching 57.7% accuracy during training, and when subjected to the same evaluation as our models, it reached an accuracy of 29.7%. This, although still significantly worse than our models, means that big, multilingual models can be trained for processing unusual language, but custom-built solutions will usually be better.

VI. IMPLEMENTATION

We have used Python for the implementation, mostly because most state-of-the-art machine learning tools are accessible as Python libraries, and it provides an easy and fast way to create a simple application that is capable of tagging texts. We have uploaded our implementation to GitHub at https://github.com/szakacsb/dictionary_expander, together with links for models and for the xml file needed to initialize the Solr database.

For storing the dictionary, we used Apache Solr[2]. Solr is a powerful search platform with a multitude of functions that made it ideal for quick lookups and storing data in a dictionary-like format. We did not utilize most of its advanced functionality, but it provided a robust out-of-the-box solution for storing data.

A. The Application

The Application itself was written purely in Python, including the parts for populating the Solr server with data, the client querying the server for data, the training and evaluation of models, and the functional tagger. The machine learning parts rely on Keras, and for the distance-based part, we used python-Levenshtein.

The application does not rely strictly on the Mikes dictionary as a corpus: using the same format, any dataset can be used to teach the model. This means that reconfiguring it for different task is fairly simple, be it dictionary expansion for a different corpus or an entirely different entity recognition and tagging task.

B. The Solr Server

While Solr is a much more robust technology than required for this task, its performance is a significant upside for this application. We transformed and uploaded the half-done dictionary and used it as our database server for the experiments.

The results, after being manually checked, can very easily be fed back into the Solr server, making further training of models possible. An incremental workflow can be created, where the application tags the text, the expert manually corrects it, and then it is uploaded into the server, and used for further training for the models.

VII. Conclusion

We have created an expert system-based automatic tagger that can be used to pre-process texts for dictionary-expansion. We have demonstrated that a three-component tool performs better on Mikes Kelemen's writings that are in an archaic dialect of the Hungarian language, and we compared our results to some already existing tools on the same corpus.

Whereas the tool we created was specifically designed for this task, it can be used in many other applications, and its flexibility allows for processing other non-contemporary or otherwise drastically different dialects.

The accuracy of the predictions is not fit for unsupervised dictionary expansion; however, we have reached a 65.9% accuracy on unknown words and this makes this tool ideal for pre-processing texts before manual corrections.

We also built the system into an easy-to-use application, together with a Solr-based server that stores the dictionary itself.

For future works we will be developing the decision-making component, using the posterior probabilities of the softmax layers and trying different, more complex approaches. We will also be looking at more sophisticated distance-based methods and more complex neural networks to try to diversify the components even further.

VIII. Acknowledgments

This work was supported by the European Regional Development Fund of the European Union under the EFOP-3.6.2-16-2017-00013 Project.

References

[1] A. Toprak and M. Turan, "English Automatic Dictionary Creation with Natural Language Processing", *2019 Innovations in Intelligent Systems and Applications Conference (ASYU)*, Izmir, Turkey, 2019, pp. 1-6 doi: 10.1109/ASYU48272.2019.8946431.

[2] X. Wang, Y. Zhang, Q. Li, X. Ren, J. Shang and J. Han, "Distantly Supervised Biomedical Named Entity Recognition with Dictionary Expansion", *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, San Diego, CA, USA, 2019, pp. 496-503 doi: 10.1109/BIBM47256.2019.8983212.

[3] Gentile A.L., Gruhl D., Ristoski P., Welch S. "Explore and Exploit. Dictionary Expansion with Human-in-the-Loop", Hitzler P. et al. (eds) *The Semantic Web. ESWC 2019. Lecture Notes in Computer Science*, vol 11503. Springer, Cham doi: 10.1007/978-3-030-21348-0_9

[4] George A. Miller, "WordNet: A Lexical Database for English", *Communications of the ACM Vol. 38*, No. 11, pp. 39-41, 1995 doi: 10.1145/219717.219748

[5] Margit Kiss, "The Digital Mikes-Dictionary", In: Tüskés Gábor; Bernard Adams; Thierry Fouilleul; Klaus Haberkamm (editor), *Transmission of Literature and Intercultural Discourse in Exile [...] The Work of Kelemen Mikes in the Context of Europen Enlightment [...]*, Bern: Peter, Lang Verlag, pp 288-297, 2012

[6] Tamás Mészáros, Margit Kiss, „The DHmine Dictionary Work-flow: Creating a Knowledge-based Author's Dictionary", *Proceedings of the XVIII EURALEX International Congress: Lexicography in Global Contexts*, pp 77-86, Jul. 2018

[7] Kiss, Margit, Mészáros, Tamás, "Rethinking the Role of Digital Author's Dictionaries in Humanities Research", Feb. 2019

[8] Ruis, F., Pathak, S., Geerdink, J., Hegeman, J. H., Seifert, C., & van Keulen, M. "Human-in-the-loop Language-agnostic Extraction of Medication Data from Highly Unstructured Electronic Health Records", *20th International Conference on Data Mining Workshops 2020* IEEE EDS, 2020

[9] Alfredo Alba, Chad DeLuca, Anna Lisa Gentile, Daniel Gruhl, Linda Kato, Chris Kau, Petar Ristoski, and Steve Welch „Identifying High Value Opportunities for Human in the Loop Lexicon Expansion", *HumBL2019. The third international workshop on Augmenting Intelligence with Bias-Aware Humans-in-the–Loop. In the Web Conference 2019 Companion volume*. ACM, New York, NY, USA, 2019. doi: 10.1145/3308560.3317305

[10] Attila Novák, "A New Form of Humor – Mapping Constraint-Based Computational Morphologies to a Finite-State Representation." in: *Proceedings of the 9th International Conference on Language Resources and Evaluation (LREC-2014)*, Reykjavík, pp. 1068–1073, 2014

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, "BERT: Pre-training of Deep Bidirecional Transformers for Language Understanding", arXiv preprint arXiv:1810.04805, Oct. 2018

[2] https://lucene.apache.org/solr/

[12] Akbik, Alan and Blythe, Duncan and Vollgraf, Roland, “Contextual String Embeddings for Sequence Labeling”, in: *Proceedings of the 27th International Conference on Computational Linguistics*, pp. 1638-1649, 2018

[13] Zhang, Wei, “Shift-invariant pattern recognition neural network and its optical architecture” in: *Proceedings of Annual Conference of the Japan Society of Applied Physics*, 1988

[14] Hochreiter, Sepp; Schmidhuber, Jürgen, “Long short-term memory” in: *Neural Computation 9 (8)*, pp 1735–1780, 1997, MIT Press
DOI: 10.1162/neco.1997.9.8.1735

[15] Alexis Conneau, Holger Schwenk, Loïc Barrault, Yann Lecun, “Very Deep Convolutional Networks for Natural Language Processing”, Jun. 2016, *arXiv preprint* arXiv:1606:01781

[16] Xiang Yu, Agnieszka Falenska, Ngoc Thang Vu, “A General-Purpose Tagger with Convolutional Neural Networks”, in: *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pp. 124-129, Sept. 2017, Copenhagen, Denmark, Association for Computational Linguistics
DOI: 10.18653/v1/W17-4118

[17] Jason P.C. Chiu, Eric Nichols, “Named Entity Recognition with Bidirectional LSTM-CNNs”, in: *Transactions of the Association for Computational Linguistics, Volume 4,* pp. 357-370, 2016
DOI: 10.1162/tacl_a_00104

[18] Shaojie Bai; J. Zico Kolter, Vladlen Koltun, “An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling”, *eprint arXiv:1803.01271*, March 2018

[19] Nemeskey, Dávid Márk, “Natural Language Processing methods for Language Modeling” PhD thesis. Eötvös Loránd University, 2020

Béla Benedek Szakács finished his BSc in computer engineering in 2018 and is currently doing his MSc studies in the same field at the Budapest University of Technology and Economics. He is a member of the Balatonfüred Student Research Group. His main field of study is machine learning and natural language processing.

Tamás Mészáros is an associate professor at the Budapest University of Technology and Economics. His research areas include intelligent agents, information retrieval and natural language processing.

IX. Appendix

The additional hyperparameters of the layers are the following:

1. Conv1D layers: padding = valid, data format = “channels_last”, dilation rate = 1, groups = 1, use bias = True, bias initializer = “zeros”, no kernel regularizer, no bias regularizer, no activity regularizer, no kernel constraints, no bias constraints.
2. Dense layers: use bias = True, bias initializer = “zeros”, no kernel regularizer, no bias regularizer, no activity regularizer, no kernel constraints, no bias constraints.
3. Bidirectional LSTM: use bias = True, recurrent initializer = “orthogonal”, bias initializer = “zeros”, unit forget bias = True, no kernel regularizer, no bias regularizer, no activity regularizer, no recurrent regularizer, no kernel constraints, no recurrent constraints, no bias constraints, no recurrent dropout, return sequences = False, return state = False, go backwards = False, stateful = False, time major = False, unroll = False.

Location, Proximity, Affinity – The key factors in FaaS

David Haja[1], Zoltan Richard Turanyi[2] and Laszlo Toka[3]

***Abstract*—The Function-as-a-Service paradigm emerged not only as a pricing technique, but also as a programming model promising to simplify developing to the cloud. Interestingly, while placing functions across hosts under the service platform is believed to be flexible, currently the available platforms pay little attention to co-locate connected functions, or data with the respective processing function in order to improve performance. Even though the local function invocation and data access might be an order of magnitude faster than their remote intra-cloud counterparts. In this paper, we therefore propose a Function-as-a-Service platform design that reaps the performance benefits of co-location. We build the platform on WebAssembly, a secure and flexible tool for efficient local function invocations, and on a distributed in-memory database, which allows arbitrary data placement. On top we advocate smart placement strategies for function executions and data, decoupled from the functions. Hence we envision good horizontal scaling of functions while keeping the experienced processing latency to that of a single machine case.**

***Index Terms*—Function as a Service, FaaS, WebAssembly, platform, runtime, co-location, performance**

I. Introduction

In recent years Function-as-a-Service (FaaS), often referred to as serverless computing, has become one of the popular paradigms in cloud computing. Numerous projects managed by companies and academic institutions offer FaaS services, such as Amazon's AWS Lambda [1], Apache OpenWhisk [2] Google Cloud Functions [3], Microsoft Azure Functions [4]. Using FaaS, developers do not need to care about resource allocation, scaling or scheduling, since the platform handles these. Most of these platforms operate with container technologies; the user's executable code is packed into a container that is instantiated when the appropriate function call request first arrives. With this relatively lightweight technology it is easy to arrange process isolation and resource provisioning.

FaaS platforms and solutions gain more and more attention in the Infocom domain, and researchers investigate its usability for the backend of e.g., tactile Internet applications, and edge computing-based applications. While its usage for telco services requires massive performance as well, the low end-to-end delay of such platforms is an absolute minimum in general. For example the deployment of a multi-party Augmented Reality application in an edge computing infrastructure with a FaaS platform on top provides excellent context for scaling ephemeral functions to the need of the users, but this use case also demonstrates the necessity for low latency in processing information shared between the users.

Even though communication is significantly faster the closer the parties are (e.g., in same data center, same rack, same server machine, same process), currently available FaaS platforms miss to co-locate entities that often communicate with each other. Furthermore, most of the available FaaS platforms suffer from cold-start latency when a new virtual machine (VM) or container has to be launched. The size of the image to mount, the programming language used, the number of libraries and dependencies all have an impact on this latency.

WebAssembly [5] (or Wasm) was announced in 2015, designed by the World Wide Web Consortium (W3C) for enabling high-performance applications inside browsers [6]. Since then a new industry partnership, the Bytecode Alliance [7], has been formed by Mozilla, Fastly, Intel, and Red Hat. The alliance's purpose is to implement standards and to propose new ones that decouple WebAssembly from JavaScript and make runtimes outside of the browser feasible. WebAssembly can essentially provide the same security capabilities and language independence as containers, but it allows for lighter composition in terms of startup and function call.

We argue that the right placement of function executions and data helps FaaS platforms to reap performance benefits of co-location. We therefore propose a novel FaaS platform design based on WebAssembly for isolation and fast local function calls; a distributed in-memory datastore to allow data mobility; and smart location selection strategies to co-locate data and function execution as the main novelty of our presented platform design. The contribution of this paper is twofold: i) we suggest to use WebAssembly as an emerging virtualization technique; ii) we consider advanced strategies for both determining the location of user functions and data. Furthermore, we define the role of each system component and major capabilities that those components need to provide. We also present the benefits, identify the challenges and missing capabilities that need to be defined for a complete system.

This paper is organized as follows. In Section II we present the related state of the art. In Section III we introduce WebAssembly, present the activity that allows WebAssembly to operate outside the browsers, highlight the benefits and the challenges of using WebAssembly in FaaS platforms. In Section IV we introduce a novel FaaS system design based on WebAssembly and locality awareness. We conclude the

[1] MTA-BME Network Softwarization Research Group, Budapest University of Technology and Economics (e-mail: haja@tmit.bme.hu)

[2] Ericsson Research, Hungary (e-mail: zoltan.turanyi@ericsson.com)

[3] MTA-BME Network Softwarization Research Group, MTA-BME Information Systems Research Group, Budapest University of Technology and Economics (e-mail: toka@tmit.bme.hu)

DOI: 10.36244/ICJ.2020.4.3

paper in Section V. All existing and missing features of WebAssembly mentioned throughout the paper reflect the status at the time of writing this paper.

II. RELATED WORK

One of the most popular FaaS computing platforms is Amazon's AWS Lambda [1]. Since the appearance of Amazon's AWS Lambda, numerous research initiatives [8]–[11] have improved their FaaS platform performance, with advanced dependency loading and caching, grouping functions related to the same application within the same container, or co-locating functions with their data. None of these works consider co-locating functions that communicate with each other and they all consider VMs or containers as virtualization technology.

Faasm [12] is a novel serverless runtime that executes distributed stateful serverless applications across a cluster. Faasm uses Faaslets, which extend the traditional WebAssembly modules with custom code that provide a minimal serverless-specific POSIX environment to support a host interface. The authors propose a two-tier state architecture: i) the local tier provides in-memory sharing; ii) the global tier supports distributed access to state across hosts. In contrast to our proposal, Faasm is a runtime implementation, it is designed to integrate with existing serverless platforms and it relies on the underlying platform's scheduler, e.g., Knative's [13] scheduler. Therefore their function scheduling strategy does not take into account the functions' behavioral patterns. Furthermore, the authors do not consider moving the data.

Cloudflare's Workers [14] service has recently started to support creating and hosting FaaS functions compiled to Wasm binaries. A novel FaaS platform is presented in [15] that uses WebAssembly and operates on an edge computing environment. AccTEE [16] is a sandbox solution that offers remote computation service with resource accounting, leveraging on two technologies: hardware-protected trusted execution environments, and WebAssembly. All of the platforms [14]–[16] are limited in the sense that they use JavaScript runtime with a wrapper JavaScript program that calls the binary instances, and returns their results. aWsm [17] is a native Wasm compiler and runtime that can operate under a serverless system. It enables Wasm shared-objects and multiple functions and their invocations within a single process.

The significant difference between all the aforementioned platforms and our design is that our design takes into account the frequency of the communication between functions and data, and strive to provide co-location for them.

III. WEBASSEMBLY FOR FAAS RUNTIME

WebAssembly defines a binary code format that is portable and efficient in size and execution speed on modern CPUs. It was originally created for executing programs in web browsers, written in languages other than JavaScript, e.g., C, C++, Rust.

A. Background

A *module* in WebAssembly [18] is the distributable, loadable and executable unit of code. WebAssembly modules are language-agnostic, which means a software developer may write its source code in any optional high-level language, such as C++, Rust, or Go, which then can be compiled, with the appropriate toolchain, to a portable binary that runs on a stack-based virtual machine.

The first technology for compiling C or C++ codes to WebAssembly binaries was Emscripten [19]. Emscripten's toolchain relies on LLVM [20] for the following key features: translating high-level code from languages (like C++) to an intermediate representation (IR), optimization and also dead code elimination. Beside Emscripten, numerous other toolchains have been developed and published for compiling other language source codes beside C and C++ to WebAssembly binaries. An example view of the compilation process is presented in Fig. 1.

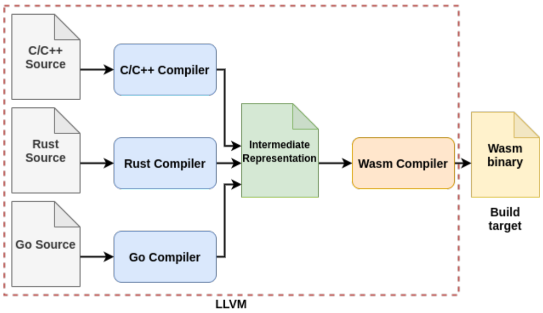


Fig. 1. Compilation to Webassembly with LLVM toolchain

One of the key features of WebAssembly is its secure execution environment. Unlike a native binary component, WebAssembly modules have access only to a part of the process memory, allowing secure isolation within a process. WebAssembly provides a sandboxed environment for the functions of a module, which by default do not have access to external APIs and system calls. To allow the interaction with anything outside the module, one has to explicitly authorize the module for the function or syscall. Taking all together, the pattern of the usage of these features is called WebAssembly "nanoprocess", which makes it possible to have similar isolation to that of a process, but with lower overhead. Another key feature is that the Wasm engine can copy directly between a caller and a callee's memories, even if they are separated in two modules and/or not compiled from the same language. This means that serialization and deserialization of communicated data may be avoided.

B. Outside the browser

WebAssembly was created as a browser runtime environment faster than JavaScript. During the compilation process Emscripten [19] created the Wasm binary and JavaScript glue code, which communicated with the browser and consequently with the API provided by the OS. This JavaScript glue code was not meant to be a standard or even a public interface. As WebAssembly became more popular and more powerful, the community realized its potential for use outside the browser. Consequently, standardization started to propose an interface, the WebAssembly System Interface (WASI) that would connect Wasm binaries to regular operating systems. WASI allows runtimes to be independent of browsers, Web APIs and JavaScript. The execution of a WebAssembly module requires a runtime that supports the standardized WASI; multiple open-source implementations of such toolchains exist, such as Wasmtime [21], Lucet [22], Wasmer [23], WAVM [24], Wasm3 [25]. These various toolchains, proposed by the community and listed in Table I, differ in their source code language, compiler framework, or compilation process.

The two most common compilation techniques are Ahead-of-time (AOT) and Just-in-time (JIT) compilation. Most of the toolchains in Table I use Cranelift [26] and LLVM as their compiler framework. Cranelift is a low-level code generator that translates a target-independent intermediate representation into various executable machine code. LLVM is a collection of modular and reusable compiler and toolchain technologies. The goal of LLVM is to provide a modern compilation strategy that is capable to support both static and dynamic compilation of arbitrary programming languages.

TABLE I
WASI COMPATIBLE TOOLCHAINS

	Source languages	Compiler framework	Compilation
Wasmtime [21]	Rust, C++	Cranelift, Lightbeam	JIT
Lucet [22]	Rust	Cranelift	AOT
Wasmer [23]	Rust, C++	Cranelift, Dynasm.rs, LLVM	JIT
Wavm [24]	C++, Python	LLVM	JIT
Wasm3 [25]	C	Custom	Interpreted

WASI is in the middle of a standardization process, but its two key design goals are already set: portability and security. In the current proposal [27], WASI consists of a modular set of standard interfaces, one of which is called *wasi-core*. Wasi-core has a similar feature set as POSIX, so it contains the very basic interfaces that functions need, like random numbers, files, network connections, etc. Although wasi-core will not implement all features of POSIX, those missing can be handled by other modules inside WASI. This way the platforms can decide, which functionality they want to use.

C. Interface types

The Minimum Viable Product (MVP) of WebAssembly [28] defines only numbers as data types. Interface types define a set of types that describe abstract, high-level types. It gives the possibility of describing complex values, e.g., strings, sequences, records, and variants, without committing to a single memory representation or sharing scheme. The complex value descriptions are mappings between multiple sets of basic types to the abstract types, where these mappings are not hardcoded in the engine, instead, a module comes with its booklet of mappings. Most of the time the compiler takes care of this information, by adding a custom section that holds the interface types, to WebAssembly modules. In cases when two Wasm modules communicate, they both give their booklets, which define how they map their functions' types to the abstract types. This allows to automatically generate code to convert between value representations of the same type in an extensible and efficient manner.

D. Benefits for FaaS platforms

Portability: Since WebAssembly is machine agnostic, the underlying operating system and processor architecture, e.g., x86, ARM, is irrelevant from the users' perspective and only one compiled binary is needed even in case of a heterogeneous server park. This portability extends even to the clients via integration of WebAssembly into the browser. This opens up the possibility of extending a FaaS system to the client.

Language agnosticism: There are numerous open-source projects [29] that make several programming languages compilable to Wasm binary. The service provider needs only one runtime implementation for all compatible source code languages, and above all, developers can enjoy the programming language of their choice. Furthermore, with WebAssembly, users can easily make function calls across languages.

Security: WebAssembly provides function invocation in a secure way by default, as the design of modules contains isolated memory and system call sandboxing, which prevents buffer overflow type of security exploits or sensitive data leaks. This, coupled with access control of the databases, makes it easier to rely on third party, untrusted code.

Low execution overhead: WebAssembly uses nanoprocesses as a virtualization technique, to provide isolation and safety during the execution of Wasm modules. This results in less computational overhead provisioned for function execution than any other technique [30].

Reduced and reliable cold-start latency: Cold-start latency is not negligible in FaaS type services. It has been shown that using Wasm modules for function invocation reduces the cold-start latency [15], so it also reduces job completion times.

Fast communication between functions: Since WebAssembly will support shared memory [31], the communication between functions on the same host can be fast and efficient, reducing serialization, value transformations to the minimum.

E. Gaps and challenges

As all technologies, WebAssembly also has its tradeoffs. Here we present the general gaps that stem from its use.

The execution of Wasm functions is slower than executing the same function natively [32]. Although the cold-start latency can be reduced significantly compared to container-based systems, the slower execution may erode that gain.

For letting Wasm modules reach host resources, like the file system, one must provide the necessary capabilities to the executor explicitly, i.e., users should provide all the capabilities and access rights that their applications require. In addition, users have to provide bindings to other Wasm modules manually.

Interface types allow modules to use more complex types than numbers. This concept is under construction, so it can change as the community moves forward. Furthermore, at the time of writing, interface types are only available for Rust programming language.

One of the biggest promises of WebAssembly is its standardized form that gives compilation target for many programming languages, although some of the source languages are not yet supported with WASI, e.g., Go [33]. Furthermore, the interpreted languages like Python, cannot be compiled to Wasm binary directly. The workaround that currently allows compiling programs written in such languages is to compile both the interpreter VM with the user's functions together in one Wasm binary. When one has many functions, this may be a significant overhead.

Currently features, like threads, exception handling, Single Instruction Multiple Data, shared memory [31], are only planned for WebAssembly [34]; their status varies between the proposal and the feature standardization phases. Some of these capabilities are indispensable features for users' functions. They are still changing and uncertain when and how they will be supported by standards.

Often one wants to execute numerous function instances in a distributed fashion, therefore the modules demand a method that provides the safe communication ability between each other. A major conceptual challenge is to provide a secure and efficient way for WebAssembly modules on different hosts to communicate through the network.

IV. Advanced WebAssembly-based FaaS design

In this section we present our envisioned FaaS system architecture: we define each component's functionality, and we emphasize the importance of the location of user functions' execution within a cluster. Throughout this paper the user denotes the application/function developer and provider that realizes the application and wishes to execute it in the platform.

A. Motivation for co-location

The main goal of our proposed system design is to enable FaaS functionality for users, with both horizontal scaling, and with small function composition and data access overhead, while retaining the flexibility and security of the currently available platforms. Our key observation is that local data access and function invocations within the same server are typically an order of magnitude faster than remote ones [35], [36]. Some illustrative overhead values can be found in Table II. Naturally the overhead depends heavily on the underlying system and network characteristics, although it is visible that they are not negligible compared to a function initialisation time. In the table we list data access times measured in DAL [35] and in S6 [37], two distributed in-memory key value stores. Also, we depict the range of function invocation latencies with Faasm [12], [17] and with gRPC [38], [39], which is a high-performance, open source universal RPC framework.

One can see that the overhead of remote operations can be heavy, especially if they have to be repeated several times. Since the novel applications and services usually require multiple instances of the same function, and numerous different functions that constitute a long pipeline, we cannot ignore the extra overhead that these operations incur. Therefore, to accomplish our performance goals, we maximize the locality effect by co-locating functions with their data, and co-locating caller and called functions.

TABLE II
FUNCTION CALL AND DATA ACCESS OVERHEAD WITHIN A DATA CENTER

	Overhead
Wasm function initialisation	$185\mu s - 5ms$
Local data access (DAL)	$1\mu s$
Remote data access (DAL)	$20\mu s$
Local data access (Redis)	$70 - 90\mu s$
Remote data access (Redis)	$70 - 90\mu s$
Local data access (S6)	$0.16\mu s$
Remote data access (S6)	$18\mu s$
Wasm - Wasm call locally (Faasm)	$235\mu s - 2.9ms$
gRPC call (between GCE VMs)	$74 - 629\mu s$
gRPC call (over Ethernet 40G)	$80\mu s - 6ms$

B. Architecture

We want to hide the distributed nature of the platform infrastructure from the application developer, and make the assumption on the cost of data access and function invocation to be in the order of what is typical in a single-machine application. WebAssembly helps this endeavor by offering low-overhead function calls between sandboxes. Our main architectural choices are: i) functions are compiled to Wasm binaries, uploaded and reachable from every worker host; ii) functions are stateless and store their data in a separate, distributed in-memory database accessible from anywhere in the system; iii) data in the database is automatically moved around to minimize access latency; iv) a function may invoke another functions locally or remotely, the choice is based on performance considerations.

We present the components of our FaaS system architecture, and a high-level overview is depicted in Figure 2.

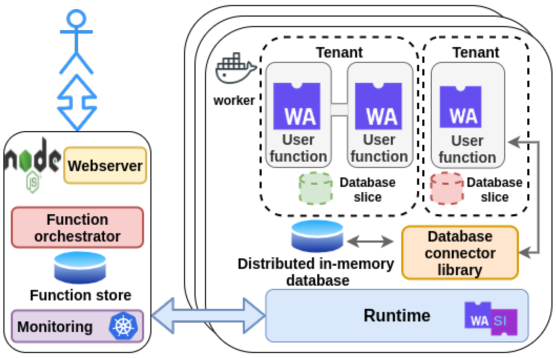


Fig. 2. Architecture of our envisioned FaaS system design

User functions: They realize the application that users wish to execute. The user can develop functions in any source language that can be compiled to Wasm binary, then the system stores only the compiled modules and executes them upon request. User functions can be invoked externally via a HTTP API, by another functions' direct calls (synchronously, asynchronously or as co-routines) or by assigning functions to trigger events, such as a change of a database item, an error (e.g., function failing) or another system event, like scaling.

Worker: Contains and manages the system components used for user functions' execution and communication. The workers can be either physical or virtual machines. We can use Kubernetes [40], as the underlying cluster infrastructure manager, where the pods represent our worker nodes.

Runtime: The WASI-compatible runtime is responsible for the execution of the User functions, memory management, sandbox creation and job scheduling. It handles code signature validation, other Wasm module dependencies and function invocations, either locally, or across different workers. It is also responsible for authorizing the sandboxes appropriately, e.g., whether functions may access certain parts of the database, files, network or some static data in a shared filesystem. We assume that all uploaded functions (and all their dependencies) are available (eventually) at every worker. This means that every worker can execute any function, allowing for execution locality for each invocation independently. For example, function *F* working on data *D1* and *D2* may be invoked at different workers depending on the location of *D1* and *D2*. Consequently, there are no resources allocated on a per function basis - most prominently functions do not get their own containers. This removes the need to manage per-function resources (e.g., scaling them) and makes it lightweight to trade resources between functions (just by making workers execute different ones). Instead, resources are managed on a per tenant (or per application) basis.

Distributed in-memory database: Since User functions are stateless, a distributed database, e.g., a key-value store, provides state sharing between function invocations. Functions may have restricted access to (parts) of the database for security and modularization purposes. The realization of this feature is crucial, as due to the stateless nature of the functions, access to the database can be a bottleneck. Databases [35], [37] that allow controlling the location of items and offer optimized local access are best suited to our architecture, as they help exploit the benefits of locality. Note that there may be more than one kind of database integrated into the system offering different semantics. Besides simple key-value stores, strongly transactional databases, conflict-free replicated data types, graph databases or any kind of distributed database can be integrated.

Database connector library: Provides connection between the User functions and the Distributed in-memory database. It offers an API directly accessible from the User functions.

Tenants: They represent the users of the system. We provide tenant isolation that prohibits users accessing binaries and data of other tenants. It also involves careful management of CPU and memory resource usage, both of the Runtime and of the Distributed in-memory database. The database slices, in terms of resource usage, are depicted with dashed line contoured database icons in Figure 2.

Function orchestrator: This manages User function execution and data locality. Since at function execution there is very little time to make a decision, a set of easy to evaluate rules must be applied to decide on the place of executions. The task of the Function orchestrator is to observe system behaviour, i.e., cross-function invocations and data access patterns, i.e., which function executions invoke what other functions, and access what data, respectively, and evolve the rules.

Webserver: Responsible for handling external requests through an HTTP API. It authorizes incoming requests, processes and invokes the appropriate User functions through the Runtime component. After the invocation, the Webserver is responsible for sending the results back to the client. The tenants may also expose (parts of) the database through this component. A general HTTP API implementation, like Node.js, is applicable for the functionality our platform requires. The performance improvement of our platform stems from the reduced communication overhead on the frequent calls between functions and between function and data. Since the Webserver does not take an active part in these communications, only in the rare user interaction, it may run on a dedicated node or on the Workers as well.

Function store: Tenants can upload and manage functions, modules, versions and aliases via this component. It is also responsible to distribute the uploaded functions to the Workers.

Monitoring: This component extends Kubernetes' cluster manager functionality with log collection, trace and alarming indicated by the User functions of a Tenant or application. It allows a unified view of the system and helps troubleshooting; it also collects performance metrics and counters.

C. Workflow

We present in Figure 3 the workflow of i) function development, upload; ii) invocation; iii) our Function orchestrator component in our envisioned FaaS platform.

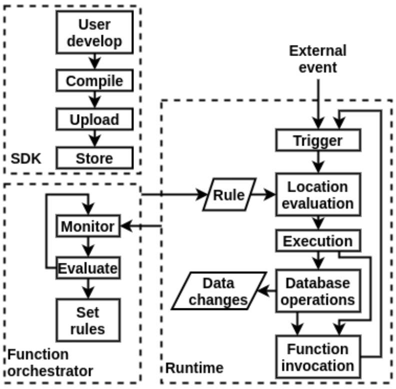


Fig. 3. Workflows of our platform

The workflow begins with the development of the user's function. As we stated before, a wider range of programming languages can be used than in the currently available FaaS platforms thanks to the WebAssembly ecosystem. After the development, either the system or the user compiles the source code to WebAssembly binary.

The binary and its dependencies will be uploaded and stored in the Function Store and distributed to all workers. Distribution may simply mean the publishing in a networked file system. When a WebAssembly module is uploaded, a URL is assigned to it to make the function externally accessible.

The users can invoke their functions via HTTP requests through the assigned endpoints; authorization is managed by the webserver. The process of a function invocation is presented with the solid lines in Figure 3.

A function running may access the database through the connector library, reading and writing data. Such writes constitute the side effect of the function and may trigger other functions if they were assigned to the change of that key. The functions themselves may also trigger other functions directly. They may also fail (or throw an unhandled exception), which may also serve as function startup trigger. In any case the runtime makes a decision on whether the new function shall be locally or remotely executed. This decision is based on a few simple rules set by the Function Orchestrator. In case of remote execution, a network message is sent to the selected worker containing the name of the function to execute and its arguments. In case of local execution a WebAssembly sandbox may be created (if isolation requires it) for the function, the arguments are moved into the sandbox and execution is passed to it. The runtime may maintain a set of pre-allocated sandboxes, for performance reasons. After local or remote function execution, if the result of the function is relevant it is transported back to the caller, either as a syncronous return value or as a promise/future, or a callback argument or a co-routine yield, depending on what construct the source programming language supports.

The Function Orchestration needs to specify the rules for execution location by observing data access and function invocation patterns. E.g., for functions that typically execute quickly and do not access the database at all, it is beneficial to always run them at the node of trigger. Similarly, functions having modest compute requirements, little database access, but large input/output data sizes are best executed locally. In contrast, functions accessing a lot of data that is remote may perform better at the remote location.

Naturally, every invocation of a function may have a different access and compute pattern, thus a single rule per function will result in many bad decisions. Not only the arguments to the function, or that of its caller can be taken into account, but metadata supplied by the caller. For example, in a telecom system, which handles a lot of users concurrently, the identity of the user can be supplied for every function execution. This allows the system to observe data access and function invocation patterns per user and group all functions working on the same user.

D. Co-location aware function scheduling

In general, function scheduling is a hard problem with multiple constraints [41]. In our platform we consider the joint orchestration of user functions and the used data. The goal is to minimize the cost that comes from data movement, remote data access and remote function calls, as all the communication through the network and data serialization-deserialization put additional overhead on the execution of user applications. Nowadays, the most widely used distributed, in-memory databases, like Riak [42], can store multiple replicas for each data (or function state), but do not allow to define the host of the state or its replica. To control data locality we need a distributed, in-memory database that provides an API, which lets the service provider define the host of each replica.

For functions that typically execute quickly, invoke each other and do not access the database at all, the best strategy is to run them at the node of the trigger (locally). Similarly, functions having modest compute requirements, rare database access, but large input/output data are best placed locally. In contrast, functions intensively accessing remote data (hosted at a node different from the node of the trigger) may perform better at the remote location, in order to access data locally there, with minimum cost.

In our envisioned system the Function Orchestrator defines rule sets that summarize its knowledge about how functions behave. To construct an adequate placement strategies, we have to identify the functions' behavioral patterns. In addition to the trivial cases there are numerous more complicated ones that need more advanced strategies. For example, we want to place a pipeline on the same host with the data that it accesses, if its functions work on the same data. To accomplish this, the pipeline needs to be decoupled and the relations between its components must be identified.

In addition, every invocation of a function may have a different access and compute pattern, thus a single rule per function will result in bad decisions. On the other hand, quite many things may influence how a function executes (such as its input parameters and the state of the context it works on) and it is not realistic to consider the state of the entire database in rules. As a compromise, we could allow the caller of the function to provide hints (such as a hash of some ID relevant to the function) that can be made part of the rules. For example, in a telco system, which handles many users concurrently, (the hash of) the identity of the user can be supplied for every function execution. This allows the system to observe data access and function invocation patterns per user and group all functions working on the same user.

Scheduling functions and data jointly implies countless open questions that we do not cover in this paper, but it is already clear that identifying the appropriate patterns is crucial for effective scheduling. Finding the right mix of programmer input and machine learning is another area of open research.

When a function is invoked, the runtime can build a directed graph that present the communications relationship between functions and data. An illustrative graph is presented on Figure 4, where $G = (F, D, E)$; $F = \{Functions\}$; $D = \{Data\}$; $E = \{f_i- > f_j; f_i- > d_k; d_k- > f_j; |\forall f_i, f_j \in F; \forall d_k \in D\}$. A node in the graph can be a data or a function instance and the directed edges present a function invocation or data access from the caller/event to the called/data. The runtime can use postorder traversal to define each entity location by applying the actual set of rules on all nodes. This constructed graph is similar to an *extended service call graph* [43].

Evaluating the rules, the runtime makes a decision on whether the new function shall be locally or remotely executed. In case of remote execution, a network message is sent to the selected worker containing the name of the function

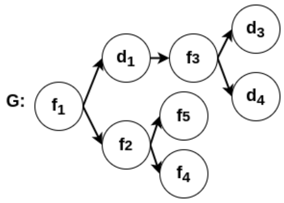


Fig. 4. Communications relationship

to execute and its arguments. In case of local execution, a WebAssembly sandbox may be created (if isolation requires it) for the function, the arguments are moved into the sandbox and execution is passed to it.

Previously, we have considered placing function executions, although data can be moved around in our system, as well. One simple strategy would be to move the data (and its replicas) to the locations with most accesses. This decouples function and data placement and considers cases in which data movement is cheaper than moving functions.

V. Conclusion

Although the appearance of WebAssembly outside the browser and of its system interface is recent, the community has already started to work on solutions that extends its usage to serve as a rapidly emerging virtualization technology under novel services. While exploiting its features of very light virtualization, available FaaS platforms do not take into account the co-location of functions, hence they neglect this potential source of performance increase. (Note that most microservice or service mesh tools also disregard locality: usually no effort is made to co-locate microservices often invoking one another.)

In this paper, we proposed a FaaS system design that offers horizontal scaling at the performance promise of host internal operation. The platform offers scaling to several server machines, when more compute power is needed. At the same time, through intelligent placement of data and function executions it minimizes the need for remote data access and remote function invocation. Not all application setups permit the reduction of remote operations under high-load, e.g., there are computing problems with mesh-like interaction of functions and data, in which case it is not possible to systematically reduce the fraction of remote operations. On the other hand, most of the computing problems exhibit clusters of often communicating functions and their data. These applications run faster in a FaaS platform considering locality.

The use of WebAssembly is instrumental in reducing the overhead of local function invocations. In addition, it offers security and portability benefits and is programmable in a wide range of languages. Although WebAssembly lacks some of the necessary features that are not implemented or defined yet, we presented how WebAssembly and its new extension WASI could fit in a FaaS system architecture. Beside the benefits, we also demonstrated the tradeoffs that will be encountered when we build the platform: the current challenges and gaps that need to be implemented before our vision becomes reality.

In our future work, we want to identify the function invocation and data access patterns based on the state-of-the-art. After the identification, we will compose and implement the Function orchestrator component that is able to construct those rule sets, and thus to improve the co-location of the functions, achieving better application performance than the available FaaS platforms.

Acknowledgments

Project no. 2018-2.1.17-TÉT-KR-2018-00012 and 135074 have been implemented with the support provided from the National Research, Development and Innovation Fund of Hungary under the 2018-2.1.17-TÉT-KR and FK_20 funding schemes.

References

[1] *AWS Lambda*, https://aws.amazon.com/lambda/, Accessed on October 25, 2020.

[2] *Apache OpenWhisk*, https://openwhisk.apache.org/, Accessed on October 25, 2020.

[3] *Google Cloud Functions*, https://cloud.google.com/functions, Accessed on October 25, 2020.

[4] *Microsoft Azure Functions*, https://azure.microsoft.com/services/functions/, Accessed on October 25, 2020.

[5] A. Haas *et al.*, "Bringing the web up to speed with WebAssembly," in *ACM SIGPLAN*, 2017. doi: 10.1145/3062341.3062363.

[6] *WebAssembly Working Group*, https://www.w3.org/wasm/, Accessed on October 25, 2020.

[7] *Bytecode Alliance*, https://bytecodealliance.org/, Accessed on October 25, 2020.

[8] S. Hendrickson *et al.*, "Serverless computation with openlambda," in *USENIX HotCloud*, 2016.

[9] I. E. Akkus *et al.*, "SAND: Towards High-Performance Serverless Computing," in *USENIX Annual Technical Conference*, 2018.

[10] E. Oakes *et al.*, "SOCK: Rapid task provisioning with serverless-optimized containers," in *USENIX Annual Technical Conference*, 2018.

[11] V. Sreekanti *et al.*, "Cloudburst: Stateful Functions-asa-Service," *arXiv preprint arXiv:2001.04592*, 2020.

[12] S. Shillaker and P. Pietzuch, *Faasm: Lightweight Isolation for Efficient Stateful Serverless Computing*, 2020. arXiv: 2002.09344 [cs.DC].

[13] *Knative: Kubernetes-based platform to deploy and manage modern serverless workloads*, https://knative.dev/, Accessed on October 25, 2020.

[14] *Cloudflare's Workers service*, https://workers.cloudflare.com/, Accessed on October 25, 2020.

[15] A. Hall and U. Ramachandran, "An execution model for serverless functions at the edge," in *Proceedings of the International Conference on Internet of Things Design and Implementation*, 2019. doi: 10.1145/3302505.3310084

[16] D. Goltzsche *et al.*, "AccTEE: A WebAssembly-based Two-way Sandbox for Trusted Resource Accounting," in *Proceedings of the 20th International Middleware Conference*, 2019. doi: 10.1145/3361525.3361541.

[17] P. K. Gadepalli *et al.*, "Challenges and Opportunities for Efficient Serverless Computing at the Edge," in *SRDS*, 2019. doi: 10.1109/SRDS47363.2019.00036.

[18] *WebAssembly Specification*, https://webassembly.github.io/spec/core/index.html, Accessed on October 25, 2020.

[19] *Emscripten toolchain*, https://emscripten.org/, Accessed on October 25, 2020.

[20] C. Lattner, "Introduction to the llvm compiler system," in *Proceedings of International Workshop on Advanced Computing and Analysis Techniques in Physics Research, Erice, Sicily, Italy*, 2008.

[21] *Wasmtime: A small and efficient runtime for WebAssembly and WASI*, https://wasmtime.dev/, Accessed on October 25, 2020.

[22] *Lucet: Fastly's native WebAssembly compiler and runtime*, https://github.com/bytecodealliance/lucet, Accessed on October 25, 2020.

[23] *Wasmer: The Universal WebAssembly Runtime supporting WASI and Emscripten*, https://github.com/wasmerio/wasmer, Accessed on October 25, 2020.

[24] *WAVM: WebAssembly virtual machine*, https://github.com/WAVM/WAVM, Accessed on October 25, 2020.

[25] W*asm3: A high performance WebAssembly interpreter written in C*, https://github.com/wasm3/wasm3, Accessed on October 25, 2020.

[26] *Cranelift Code Generator*, https://github.com/bytecodealliance/cranelift, Accessed on October 25, 2020.

[27] *WASI: WebAssembly System Interface*, https://github.com/bytecodealliance/wasmtime/blob/master/docs/WASI-overview.md, Accessed on October 25, 2020.

[28] *WebAssembly Minimum Viable Product*, https://webassembly.org/docs/mvp/, Accessed on October 25, 2020.

[29] *Awesome WebAssembly Languages*, https://github.com/appcypher/awesome-wasm-langs, Accessed on October 25, 2020.

[30] *Building a secure by default, composable future for WebAssembly*, https://hacks.mozilla.org/2019/11/announcing-the-bytecode-alliance/, Accessed on October 25, 2020.

[31] *Multi Memory Proposal for WebAssembly*, https://github.com/WebAssembly/multi-memory, Accessed on October 25, 2020.

[32] A. Jangda *et al*., "Not so fast: analyzing the performance of webassembly vs. native code," in *USENIX Annual Technical Conference*, 2019.

[33] *wasm: support new WASI interface*, https://github.com/golang/go/issues/31105, Accessed on October 25, 2020.

[34] *WebAssembly Features to add after the MVP*, https://webassembly.org/docs/future-features/, Accessed on October 25, 2020.

[35] G. Németh *et al*., "DAL: A Locality-Optimizing Distributed Shared Memory System," in *USENIX Hot-Cloud*, 2017.

[36] M. Szalay *et al*., "Industrial-Scale Stateless Network Functions," in *IEEE CLOUD*, 2019. DOI: 10.1109/CLOUD.2019.00068.

[37] S. Woo *et al*., "Elastic scaling of stateful network functions," in *USENIX NSDI*, 2018.

[38] *gRPC Benchmarking*, https://grpc.io/docs/guides/benchmarking/, Accessed on October 25, 2020.

[39] R. Biswas *et al*., "Designing a micro-benchmark suite to evaluate gRPC for TensorFlow: Early experiences," *arXiv preprint arXiv:1804.01138*, 2018.

[40] *Kubernetes: Production-Grade Container Orchestration*, https://kubernetes.io, Accessed on October 25, 2020.

[41] E. Van Eyk *et al*., "A SPEC RG cloud group's vision on the performance challenges of FaaS cloud architectures," in *Companion of the ACM/SPEC International Conference on Performance Engineering*, 2018. DOI: 10.1145/3185768.3186308.

[42] *Riak: Enterprise NoSQL Database*, https://riak.com/, Accessed on October 25, 2020.

[43] M. Obetz *et al*., "Static Call Graph Construction in AWS Lambda Serverless Applications," in *USENIX HotCloud*, 2019.

Dávid Haja is a Ph.D. student at Budapest University of Technology and Economics. He is a member of the High Speed Networks Laboratory (http://hsnlab.hu) at the Department of Telecommunications and Media Informatics. His main research interests include Edge Computing, Software-Defined Networking (SDN), Network Function Virtualization (NFV) and Resource Orchestration.

Zoltán Richárd Turányi received his M.Sc. degree in Computer Science from Budapest University of Technology and Economics in 1996. In 1997 he joined Ericsson's Traffic Analysis and Network Performance Laboratory (Traffic Lab). Since then he worked with various Mobile Core Network, Software Defined Networking and Network Function Virtualization projects within Ericsson research. Since 2014 he fills the role of 5G Network Architectures Expert within Ericsson Research.

László Toka is assistant professor at Budapest University of Technology and Economics, vice-head of HSNLab (http://hsnlab.hu), and member of both the MTA-BME Network Softwarization and the MTABME Information Systems Research Groups. He obtained his Ph.D. degree from Telecom ParisTech in 2011, he worked at Ericsson Research between 2011 and 2014. His research focuses on cloud computing and artificial intelligence.

Supporting Digital Supply Chains by IoT Frameworks: Collaboration, Control, Combination

Dániel Kozma and Pál Varga

Abstract—**The purpose of this paper is to introduce a technology-oriented SCM (Supply Chain Management) development methodology, which can be used in the design of IoT (Internet of Things) frameworks especially characterized by supply chain processes. In order to meet DSC (Digital Supply Chain) expectations, two areas are examined in detail during the literature review. Firstly, the current SCM models are studied. Secondly, Industry 4.0 requirements had to be surveyed. As a consequence, challenges and gaps are identified for which we seek the solution during our research. Based on the results, it can be stated that digitization has definitely required an improved technological solution that IoT frameworks can provide. The result is a technology-driven, IoT-based SCM development methodology that serves as a basis for the design of such platforms, which will manage supply chains. To prove the feasibility of the proposed development methodology, the Arrowhead industrial IoT framework is used for validation.**

Index Terms—**Industry 4.0, Supply Chain Management, Digital Supply Chains, Productive 4.0, Arrowhead, IoT Systems, Development Methodology**

I. Introduction

The processes of digitization and automation are setting a stronger pace than ever before, and they have a significant effect on the dynamic development of the industrial domain. The fourth industrial revolution – so-called Industry 4.0 [1] – also poses challenges for organizational reformation, and it has a significant impact on corporate and production processes. The goal is nothing else but to increase productivity and efficiency – the organizations would like to produce more with their currently available resources – which is challenging to reach without significant investments. Taking into account the growing expectations for modern supply chains, there is a clear need for technological innovation in which IoT (Internet of Things) systems can help the supply chains to adapt to new circumstances, and ultimately to transform the original supply chain into DSC (Digital Supply Chain) [2]. However, a methodology specifically for this purpose has not yet been developed. In order to define the suitable solution, it is necessary to get to know the current SCM (Supply Chain Management) processes and used models, taking into account the Industry 4.0 expectations and – based on the results – to examine what technological processes, tools, and systems can be used to provide with the right result. Matching the SCM models with the expectations of Industry 4.0, as a consequence, the gaps become visible. Supplementing the knowledge gained during a thorough literature review with empirical results, the paper's novelty strives:

1) to cover the described technology-related SCM gaps;
2) to propose a technology-driven, IoT-based SCM (hereafter IoT-SCM) framework development methodology specifically targeting the IoT domain.

The validation of the proposed IoT-SCM framework development methodology is also presented on a real industrial IoT framework, Arrowhead. Based on the results, the rule-set of the proposed IoT-SCM methodology can be used to create and manage supply chains with the help of IoT systems.

The rest of the paper is organized as follows. Section II reviews the literature on the SCM and Industry 4.0 fields; Section III examines the current challenges and research gaps of DSCs. As the main contribution, Section IV covers the identified gaps by introducing an IoT-SCM framework and platform development methodology, which is validated in Section V, using the Arrowhead framework. Section VI concludes the paper.

II. Literature Review

Supply chains are expected to be digitized over several iterations. Initially, it is easier to digitize technology-based processes such as production or logistics; however, there are parts, e.g., tendering, contracting, and other financial decisions, which will certainly be even human-driven for a while, and these areas will be suitable to rely more and more on technology and automated solutions in a later phase. Consequently, implementing the Industry 4.0 requirements is already having a big impact on the production and logistics, which leads to the management of a smart, efficiency-oriented, and value-driven supply chain.

To understand the motivation behind this technological change, it is necessary to examine how traditional supply chains are structured and function; furthermore, align this operation with Industry 4.0 expectations. This can reveal how the traditional, manual, mainly human-operated supply chains can transform into DSCs.

A. Managing Supply Chains

Basically two cross-functional, cross-firm and process-based SCM approaches exist: the SCM Framework [3], [4] and the SCOR (Supply Chain Operations Reference) [5], [6] model.

Dept. of Telecommunications and Media Informatics, Budapest University of Technology and Economics, Budapest, Hungary
E-mail: {kozma, pvarga}@tmit.bme.hu)

DOI: 10.36244/ICJ.2020.4.4

1) The SCM Framework: In this terminology, the main focus is on relationship management, which is broad in its scope, including activities such as product development, demand generation, relationship management, and returns avoidance. This breadth is why the participation of all the functional areas is critical in the SCM Framework. The model defines eight processes that touch all aspects of managing the business. Each process team is comprised of managers from all business functions, including marketing, sales, finance, production, purchasing, logistics, research, and development, as Fig. 1 shows.

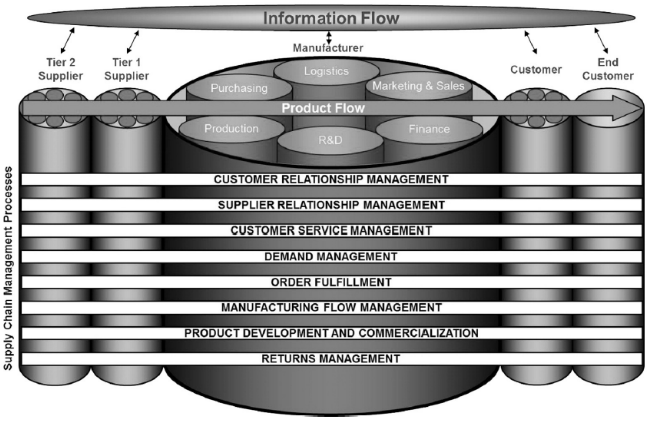


Fig. 1: SCM Framework [4]

2) The SCOR model: It is a process model of an industry-independent approach for supply chains. The model enables SCM methods to be shared and co-developed in the supply chain according to the parties' needs. The SCOR can be recognized as a reference model that can be used as a starting point for supply chain planning or development across the border of different companies. It can be used as an analysis and optimization tool for identifying the weaknesses of existing models. SCOR provides three-levels of process detail and optionally another one for company-specific considerations. At this level, the model provides an opportunity that companies define practices and individual SCM methods to attain competitive benefits and adjust to changing business conditions, as Fig. 2 shows. The SCOR model visions four main fields along with the proper SCM can be realized:

- Processes: Standard descriptions of management processes and process relationships;
- Performance: Standard metrics to describe process performance and define strategic goals;
- Practices: Management practices that produce significantly better process performance;
- People: Standard definitions for skills required to perform supply chain processes.

3) Comparison of the approaches: Based on a thorough review [4], the SCM Framework and SCOR are similar because both of them support cross-functional involvement and also realize that company functions cannot be replaced by business processes. However, the number of functions included in each framework is different, and the type of cross-functional involvement differs. In the case of SCOR, the cross-functional involvement is pursued primarily within three functions: logistics, production, and purchasing. While SCOR includes enabling as a process, the activities included resembling the SCM Framework's management components. Focusing on just three functions might make SCOR easier to implement, but this means that other areas that affect the supply chain such as research and development, marketing, or finance will be less emphasized - which, in contrast, are addressed by the SCM Framework.

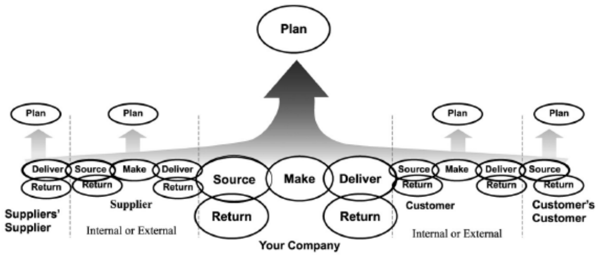


Fig. 2: SCOR model [6]

B. The Revolution of Industry

Changes in the industry affect not only manufacturers but also those processes that support production. Based on the literature [1], [7], the definition of Industry 4.0 can be summarized as a general terminology and concept for the fully-digitized production. In the modular Smart Factories of Industry 4.0, various Cyber-Physical System (CPS) processes can be monitored through virtual mapping and decentralized real-process decisions. Real-time communication and collaboration can take place between people and machines or between either of them. With the support of the IoT domain, internal and external organizational services can be provided and accessed. According to the terminology, the general requirements of Industry 4.0 are focused on the following terms.

1) Interoperability: In the Smart Factory concept, all CPSs within the factory can communicate with each other through a well-defined interface – from the workpiece carriers, through the assembly stations to the handling of products. This means the interconnection and collaboration of different machines, vehicles, equipment, and people through IoT.

2) Virtualization: The virtual mapping of the factory means that CPSs are monitored based on real-time data coming from different sensors. The data extracted from the sensors are linked to the factory's virtual model and various simulation models, whereby the physical world can be mapped and represented virtually. For a factory, the virtual model includes the status of all CPSs what helps the operators to react to different, unexpected events instantly.

3) Decentralization: The demand for individual products is increasing, and defining the core manufacturing processes is getting more and more difficult. The CPS approach allows making independent decisions in certain situations where intervention is required from a higher entity than the physical system itself, such as in cases of failures or unexpected events. In predefined manufacturing, services provided by machines can identify production workflows. Therefore, instead of fully centralized and controlled production, decentralized production can occur, which requires a well-defined production framework.

4) Real-Time Capability: This is required since operations have been drastically accelerated in various segments of our world – including industrial production and logistics. To support different organizational processes, it is essential to collect and analyze data in real-time. Industry 4.0-capable systems can continuously monitor and examine the state of the plants and the production. Thus, if a problem arises during production, the factory can respond in real-time and redirect the manufacturing process to another internal workstation or even another external factory.

5) Service Orientation: Another crucial factor concerning the feasibility of IoT and Industry 4.0 is the approach of SOA (Service-Oriented Architecture) [8]. Services provided by companies, CPSs, and people are available over digitized infrastructures and can be used by other participants or offered inside and outside the company. As a result, product-specific operations can be implemented, e.g., by the RFID tag, and can be adapted to customer-specific requirements, independently from factories.

6) Modularity: This term suggests that a system can flexibly adapt to the changes of the modular palette in case of swapping, removing, or expanding modules (or systems). In Industry 4.0, based on standard software and hardware interfaces, new modules can be automatically identified and used immediately.

III. Survey on the challenges of Digital Supply Chains

The emphasis of the DSC is not on whether raw materials, goods, products or services are digital or not; the point is that the management of supply chain processes will be digitized, using new technologies such as crewless vehicles and cloud computing, among others [2]. The more stakeholders who use digital solutions, the easier it is to drive information transparency, track goods, and optimize. As the supply chain becomes more distributed, digital transformation and stakeholder integration in a common SCM platform can bring benefits. Combining this with emerging SCM-related technologies, great opportunities can be provided for businesses to understand the supply chain better, reduce costs, and increase the speed, quality, and flow of goods.

The increase in same and next-day delivery has created a level of demand that puts new pressure on businesses. As demand grows, the entire supply chain evolves rapidly from a functional orientation to global and interconnected networks. There are many new and emerging ways for companies to reshape their supply chain to meet the needs of modern customers. Advanced technologies such as artificial intelligence (AI) [9], blockchain [10], and in general, automation [11] are being integrated into the DSCs, integrating data and any other resources from different locations to drive the distribution of manufactured goods along the value chain.

Of the two SCM models introduced earlier, the SCM Framework is complete in the sense that it also covers management processes, while the SCOR is much more concerned with logistics, production, and purchasing. However, as has been emphasized before, certain decisions cannot be brought fully automatically, such as tendering, contracting, and other financial decisions, where the human factor still plays a major role due to specific agreements and bargains. Nevertheless, the SCOR approach's scope aims to cover exactly those areas (logistics and production) in more detail, which can be automated on the technological levels. It should be noted that with the help of smart contracts [12], purchasing has already moved in the automated direction, but here the human factor is still significant.

Considering the scope, we analyze SCM's challenges according to the SCOR approach – due to its more technical nature. Therefore, the following subsections will review the challenges from the SCOR-defined fields, i.e., process-, performance-, practice- and people-related points of view.

A. Process-related challenges

1) Market growth: The cost of production is significant from research through development to product introduction. Therefore, the primary purpose of companies is to expand the emerging markets and to increase profits. Access to new market areas is difficult, especially when it comes to foreign countries, as trading policies, different fees, and government regulations may vary. The number of suppliers is also expanding. In this ever-expanding market, it is also challenging [13] to plan the supply chain in the beginning. The challenge is to find the appropriate supplier with the proper solution. Another challenging factor is reaching new customers. A further result of market growth, the efficiency of the supply chain is even more dependent on the proper tracking, authenticating the sending, receiving, and location of goods, which is a big challenge. The lack of these can lead to unnecessary delays and holdups, damaging the operations throughout the supply chain. Periodic evaluations and redesigns are required to be efficient and effective as possible. These adjustments respond to market changes — changes such as the introduction of new products, global sourcing, the availability of credit, and the need to protect intellectual property. These risks need to be identified and quantified to allow for control and mitigation.

2) Individual needs: The modern supply chains are complex. If we add products that are continually changing, the challenge is even higher [14]. The customers are putting continued pressure on companies for newer innovations, which allow them to stay competitive. In order to improve the products, companies need to transform their supply network. To respond to market needs, companies should adapt the production flow according to the needs of customers and grant a broader scope of activities, which in most cases, involves modifying the established processes. The time for introducing new products – and thus the time for manufacturing itself is decreasing. This is an increasing challenge for manufacturers, as high-quality products must continue to be guaranteed in lesser time. While expanding the product range, there is another pressure on manufacturers to produce high-quality products that are secure and safe. Untrustworthy or defective products can ruin the company's reputation, which can only be slowly, but in most cases, never fully recovered.

3) Globalization: Reducing costs is one of the biggest challenges for organizations. Therefore, to reduce the price, companies have moved to manufacture in countries where production costs are lower. However, customers want to get their products not only cheap but in time. Although production will be cheaper this way, cross-border transportation is needed, which can lead to delays; even a small decline in supply chain efficiency can have a significant impact on productivity and profitability. Without accurate status reports, supply chains can lose resources and time, making it more difficult to meet with the preplanned goals. In order to support globalisation [15], there is a need for an effective global procurement network that is capable of fully supporting the supply chain and responding to its needs in time. The careful selection of a strategic supplier is also outstanding. It can ensure the appropriate manufacturing locations and various production-supporting services on high quality. Consequently, in the globalization of manufacturing operations is important to have a global supply chain network that can effectively support and quasi in real-time response to the needs of the supply chain.

B. Performance-related challenges

1) Analytics of supply chain: According to the requirement of supply chains to be adaptable and react quickly to unexpected events or changed needs, real-time processing of data generated in the supply network is essential. Collecting the growing amount of supply chain data is not really a problem, but its effective analysis is [16].

2) Cost-control: Operating costs are under immense pressure from rapid changes in various factors such as rising material and energy prices, fuel and freight costs, fast technological changes, new regulations, and more global customers [17].

C. Practise-related challenges

1) Lack of Transparency: Traditional supply chains are often non-transparent and complex to understand. This makes it difficult to track and plan how goods and resources move through the supply chain [18]. Without complete transparency the optimization and efficient management are nearly impossible. This problem is exacerbated by older software and system solutions that do not allow information to be collected, managed, and transmitted efficiently.

2) Trusted resource exchange: Without trust, resources and information cannot be shared, and thus supply chain managed effectively. With the spread of digitization, there is an increasing need to rely on technological solutions, which poses new risks. A well-designed system provides plannable and predictable results as well as better supplier and partner relationships [19].

D. People-related challenges

1) Stakeholder integration: The consolidation of stakeholders can bring many advantages, and it can eliminate supply base variances and overheads [20], among others. Here, the challenge is to find a supplier who has the most efficient solution for the given problem. However, supply chains need to be as fast as possible to meet the demands of modern consumers. A complex, legacy supply chain often relies on the goodwill and established norms to work well. When these areas are challenged by increased demand or external disruption, relationships can suffer, together with the quality and timeliness of supplying products. It is a must to create, understand, and follow commonly agreed standards to better understand current performance and opportunities for improvement. This is an area that can become automated in a well-controlled and planned environment.

2) Governance and regulation-related challenges: A big challenge in the implementation of DSCs is to involve the massive and growing volumes of data produced today, and the tension between protecting internal data while still sharing product and consumer information with partners across distributed supply chains [21]. This is especially true for personal and sensitive personal data. For their protection, multi-government regulations have been established, and one of the best known of them is the GDPR (General Data Protection Regulation) [22]. The goal of GDPR is to protect the privacy of EU citizens expressly. While many firms believe that GDPR does not affect them because they are outside of Europe or do not directly handle customer data, in most cases, this is not true. Supply chains often cross borders and continents. Therefore, it is very important to ensure legal-compliance within these supply chains in all cases. As a result, companies must take special care when transmitting customers' or even suppliers' personal data. Besides, it can be stated that non-compliance with the regulations can lead to a huge, and in some cases, fatal monetary loss. Following the example of GDPR, damages for non-compliance could amount to €20 million or a maximum of 4% annual worldwide turnover [22].

E. Research gaps related to digital supply chains

Based on the previously-presented challenges and taking into account a comprehensive, supply chain-related research presented by [2]; the following subsections will address the current gaps.

1) Lack of development frameworks: Very incomplete, quasi no methodology is provided for digital SCM. These principles would help managers and developers to build, deploy, and use digital platforms specifically designed for DSC-related processes.

2) Lack of technology: DSCs are different from the currently widespread supply chains. Decisions in the DSC-context require new tools and technologies that take into account the digitization environment. DSC will affect maintenance, quality, inventory management, logistics, production planning, and procurement, among other issues, where there must be a system capable of bringing intelligent decisions; analyzing big data; transferring information and resource in an automated way; modeling digital twin; maintaining cybersecurity, and provide with modularity and flexibility [23].

3) Lack of integration: There are numerous barriers to the rapid implementation of DSC from both managerial and technological perspectives. Organizations are at the edge of competition to transform their supply chains digitally. Thus, technology-related expectations addressed in the previous subsections need to be resolved and implemented; furthermore, integrated in real-time. There are only a few studies on how to deal effectively with the transition from traditional supply chains to DSCs.

In the next section, a methodology will be introduced in line with the gaps that can help build DSC-compliant IoT platforms. The proposed methodology takes into account the main technological areas and also addresses integration issues.

IV. SCM-specific Design Methodology for IoT Frameworks

Based on the challenges presented in the previous sections, there is a clear need for a standard or guidance with which those needs can be served. Although many gaps have been identified in many articles, there is still no precise technical recommendation specifically describing how DSC could be supported. However, the presented gaps provide great help to define the exact requirements. The following subsections present the main pillars of our proposal on which the technical solution can be based. In some cases, there were already recommendations in the literature; still, they no longer fully meet the new requirements. Therefore, in addition to the novelties, we also supplemented the previously defined terminologies and processes, making them Industry 4.0 and DSC compliant.

A. Basic architecture

According to Sampson and Froehle [24], the basics of manufacturing supply chains are very similar to each other. As Fig. 3 shows, the traditional supply chain was a one-way, unidirectional chain where consumers may have an impact on product design, but they are mostly out of the production.

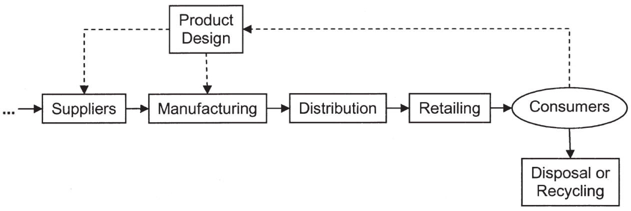


Fig. 3: Unidirectional supply chain – Traditional approach [24]

As mentioned in the challenges, on the one hand, nowadays, it is becoming increasingly crucial for the modern consumer to get individual products. On the other hand, the shortness of time from order to delivery is an increasingly critical factor. Traditional supply chains cannot serve this expectation. Here came the realization that the unidirectional chain needs to be made bidirectional. In response to this gap, the UST (Unified Services Theory) [24], [25] came alive, where the supply chain consumers have an expanded role; thus, the traditional one-way chains become bidirectional. This approach divides the supply chain roles into two major groups of stakeholders, and it led to the service provider–consumer model, as shown in Figure 4.

Based on the model, the provider can be a supplier for any resource or service requested by the consumer. This approach is the same as the service-oriented expectation advocated by Industry 4.0; therefore, the architecture for IoT-SCM design will be SOA; thus, at the same time, the bidirectional supply chain concept and the Industry 4.0 expectation are met.

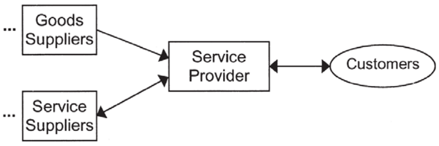


Fig. 4: Bidirectional supply chain - Modern approach [24]

In relation to SCM, three major dimensions have been identified following the literature review where the supply chains can be supported on technology level: Collaboration [26]–[28]; Combination (and integration) [29]–[31]; and Control [32]–[34]. In the following, the main activities will be introduced; besides, their further sub-activities will be defined.

The concept of CCC (Combination, Collaboration, Control) [35] can be seen in Figure 5. This model summarizes the technological requirements for developing a DSC-compliant IoT-SCM system.

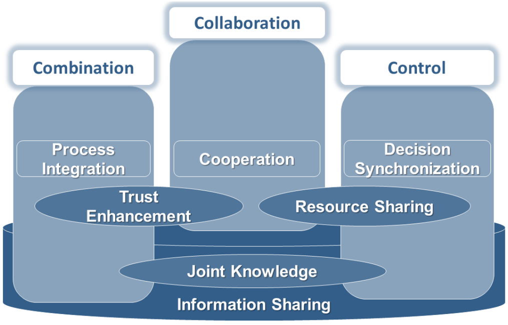


Fig. 5: Combination, Collaboration, Control

B. Collaboration - a main activity

Based on the presented research [26]–[28] and experimental results, it can be stated that it is one of the most important pillars within the supply chain in achieving effective production.

Collaboration ensures communication between participants of the supply chain and allows for the human-CPS partnership. Logistics processes, stakeholders, and production management require a well designed SCM [36]–[38]. Developing the supply chain network is always project and domain-specific, but the foundations are still similar. In this activity, the focus is on the relationships between suppliers and manufacturers, companies, and manufacturers, and last but not least, between companies and consumers, where there is a provider-consumer relationship between everyone. In order to create an effective collaboration, it is necessary to make cooperation technologically available. The verb cooperation is often used as a synonym for collaboration, but there is a significant difference between them. The target is the same, but the approach is not. Cooperation means that every participant is working to achieve the primary goal together with their own benefits; in this aspect, the collaboration combines cooperative activities for a greater purpose. Another important aspect of the collaboration is that the partners equitably distribute the risks, costs, and rewards of the production within the chain [39]. To aim this, frequent, balanced, bidirectional, and multilevel information sharing is needed that indicates a close relationship between the partners [40].

Furthermore, during collaboration, it is necessary to share not only information but physical resources as well; however, the information and resource sharing are impossible without trust. Trust is the foundation of every high-performance supply chain, and transparency is a critical element of building trust between members of the chain [41]. Trust helps contact between the participants, resulting in long-term collaboration and greater benefits. However, trust also includes providing proper security for communication, data, and different artifacts created by the chain. Based on these, the *Collaboration* main activity can be divided into four sub-activities, i.e., cooperation, information sharing, resource sharing, and trust enhancement.

C. Combination - a main activity

The term ”combination” in this context refers to the integration of several continuous or similar, independently executed processes, or any kind of resources [29]–[31], and it places more emphasis on central planning, ownership, which is governed by contractions.

The combination of supply chain processes means that the company and its partners work in sync, based on common terminology and standards, to achieve the specified business goals through integrated business processes and information sharing. To accomplish this, global processes must be inserted into the supply chain [42]; furthermore, all supply chain members must apply the same principles, standards, and procedures to reduce the risks and improve the interaction between supply chain partners and corporate strategy [4]. In this case, of course, it is also necessary to ensure trust as well. The chain members need to know that they can trust each other in every circumstance, knowing that while doing everything they do, the rest of the chain will act similarly.

Besides, processes in the supply chain must be continuously monitored and optimized. This activity will provide the participants with a better understanding of the market and the competitive environment. This will result in a joint knowledge [43], which can help future production thanks to the continuously shared information and documented experiences. Based on these, the *Combination* main activity can be divided into four sub-activities, i.e., process integration, joint knowledge, trust enhancement, and information sharing.

D. Control - a main activity

In the past, ”coordination” was the term used conceptually [44]–[46], but according to the changes and the presented expectations, now the coordination in itself has not been proven expressive enough. It is an essential part of SCM but can be subject to a more significant term, the *Control* [32]–[34]. The participants and the processes need to be coordinated and controlled from managing inventories through production management or different quality management to plan and implement considerations.

In this aspect, it can be stated that the synchronization of decision-making [47] is an important control-related task. The decision synchronization is a must between participants in a DSC; the common design, implementation, or process-related questions have to be clarified. Furthermore, market decisions such as pricing, decisions about various product features, or the optimal order quantity regarding materials or products are decision-based factors as well. Nevertheless, it also has a critical issue that not everyone has the right to make decisions at all levels, which is leading to conflicts. To avoid this situation, a transparent framework, clear roles, and coordination are needed [48].

Continuing this line of reasoning, the flow of information must also be directed and controlled. It is important that participants have access to information based on privileges and that unauthorized people do not have access to business secrets. The establishment and maintenance of information security [49]–[52] is an essential condition for next-generation systems, partly because they make it easier to comply with legal rules such as GDPR. It follows from the management of information sharing that the knowledge achieved during the management of supply chain processes cannot be made available to unauthorized people either. This kind of knowledge usually provides a competitive advantage to the actors in a given supply chain over the competition.

In DSCs, resource sharing becomes dynamic, in some cases fully automated; for example, on the production level, robots are automatically allocated to production lines, or even a production process is dynamically taken over by another factory. Based on these, the *Control* main activity can be divided into four sub-activities, i.e., decision synchronization, information sharing, resource sharing, and joint knowledge.

E. System classification

Based on the CCC model defined in the previous sections, it is possible to classify the planned IoT-SCM system's subsystems. According to the sub-activities described in the CCC model, it is possible to determine what activities a subsystem is involved in, and the importance of the system can be determined accordingly. In the CCC, seven sub-activities are identified.

1) Grade I.: If a subsystem is involved in a minimum of six out of seven sub-activities of CCC, it will certainly be an indispensable subsystem from the entire system point of view. All stakeholders need to implement and use it; therefore, it can be considered as a mandatory subsystem, in this terminology, a Grade I. system.

2) Grade II.: If a system is involved in three to five sub-activities of CCC, then it can be considered as a less important subsystem, which is necessary for a specific actor, or actors of the supply chain, but not globally obligatory from the whole supply chain point of view. Without them, the supply chain can now work together, but perhaps not as effectively; therefore, it can be considered as a supporting subsystem, in this terminology, a Grade II. system.

TABLE I: System classification

Sub activities [number]	Grade classification
6-7	Grade I.
3-5	Grade II.
1-2	Grade III.

3) Grade III.: If a system is involved in one to two sub-activities of the CCC model, then that system is specifically important to only one actor, mostly some kind of local system or application that is solely responsible for a particular sub-activity. From a global perspective, it is among the least dependable subsystems, in this terminology, a Grade III. system.

According to the classification method, the Table I summarizes the suggested Grades in relation to the sub-activity numbers. Naturally, it can depend on individual decisions of the given context, from which it follows that the borders may change.

This kind of classification is important because, on the one hand, the cross-operation within the supply chain requires that actors use the same platform where it is a must to know what systems are required to enable participants to communicate and collaborate. On the other hand, reacting to the new trends, new subsystems will be developed to meet changing expectations. This classification technique also helps to classify and integrate new systems into the existing ecosystem, with the appropriate weighting.

F. Lifecycle management and toolchains

The modular approach of Industry 4.0 requires systems to be integrated into industrial systems (e.g., CPS) as efficiently and quickly as possible. This requires – among others – up-to-date knowledge of the system states and the actions needed at state transitions, and its success depends on the consistent development and management of systems through their lifecycle. Whereas systems in the industrial ecosystem are constantly changing, new ones are integrated, old ones are taken out; therefore, these actions must dynamically take place, along with preplanned processes [53]. To this end, for the IoT-SCM framework, a lifecycle model should also be defined to support these procedures, as well as a toolchain in line with the lifecycle model. Therefore, all supply chain actors are aware of the need for such tools for the given IoT-SCM system. Special care must be taken in the lifecycle and toolchain management [54], [55]:

- to meet the Industry 4.0 requirements;
- to accommodate standard modeling techniques;
- to provide a single framework for the design, implementation, integration, and management of systems;
- to maintain the hierarchical and modularity expectations;
- to provide tools for the agile and dynamic construction of systems;
- to enable the automatic deployment of systems;
- to enable interoperability and integrability for heterogeneous systems;
- to enable a service-oriented architecture guaranteeing adaptable, loosely coupled, and late-bound services;

furthermore, which can be measured, evaluated, and can adjust to new trends and changes.

V. Validation of the IoT-SCM Development Methodology

The concept of the development methodology described in the previous sections needs to be validated on a minimum prototype implementation. For this purpose, the Arrowhead industrial IoT framework will be used. The Arrowhead [56] is an open-source project developed by ARTEMIS – *Advanced Research and Technology for Embedded Intelligence Systems* – the European Technology Platform for Embedded Computing Systems. Now, Arrowhead has been further developed within ECSEL Productive 4.0, which is an ambitious holistic innovation project that aims to open up the opportunities for the Digital Industry.

A. Basic Architecture of Arrowhead

In line with Industry 4.0 expectations, Arrowhead is also based on the SOA. The framework envisions local automation clouds capable of performing both local and remote tasks, creating dynamic collaboration between different participants. Besides, according to the other requirements of Industry 4.0 (i.e., flexibility and modularity), it supports the collaboration of both legacy and newly built CPS architectures. Similar to the UST concept (see in Figure 4), Arrowhead separates service providers and consumers, as shown in Figure 6, in line with SOA. The figure also shows the Arrowhead systems (detailed in the next subsection) that all participants must implement to use services from each other.

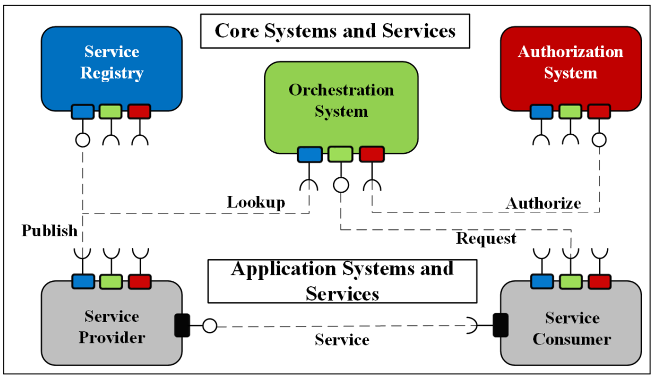


Fig. 6: Service-oriented approach of Arrowhead [56]

B. Arrowhead Systems and Classification

The current Arrowhead systems and their classification are illustrated in Figure 7. As shown in Figure 6, in order for the service-oriented operation to be realized, there are three mandatory, Grade I. systems.

1) Grade I. systems: In the Arrowhead terminology, these are the so-called Mandatory Core Systems, which includes the three basic pillars of the framework [56]:

- Orchestration System is a central component of Arrowhead. The process of orchestration is essential in support of service re-usability, service discoverability, and service composability. From an architectural perspective, the Orchestration System is responsible for finding and pairing service consumers and providers;

- Service Registry provides storage of all active services registered within a local cloud and enables the discovery of them for remote clouds too. A local cloud should contain only one Service Registry. Here are all three important elements of SOA that are fulfilled: loose coupling, late binding, and lookup;
- Authorization System provides Authentication, Authorisation, and optionally Accounting (AAA) of a system consuming a produced service.

2) Grade II. systems: In the Arrowhead terminology, these are the Supporting Systems, which help to create and operate SoS (System of Systems). The set of Supporting Systems is growing, as new ones appear according to current expectations. They are involved in managing more sub-activities defined in the CCC model. Such systems include QoS (Quality of Service) Manager [56], System Configuration Store [56], Gatekeeper System and Gateway [57], Event Handler [58], Plant Description System [59], Translator System [60], Historian [56] and Workflow Choreographer [54], [61].

3) Grade III. systems: Alternatively, in Arrowhead, these are the Local Cloud Specific Systems or Application Systems, which are part of a CPS with sensory and functional capabilities in the ”real world”. Arrowhead does not make any assumptions about what an Application System might be. It can be a single sensor or a whole, large smart environment. The emphasis here is rather on the fact that an Application System provides and consumes services from the other local cloud systems, and Mandatory Core Systems govern this information exchange. For the most part, they perform a single, specific sub-activity defined in the CCC model.

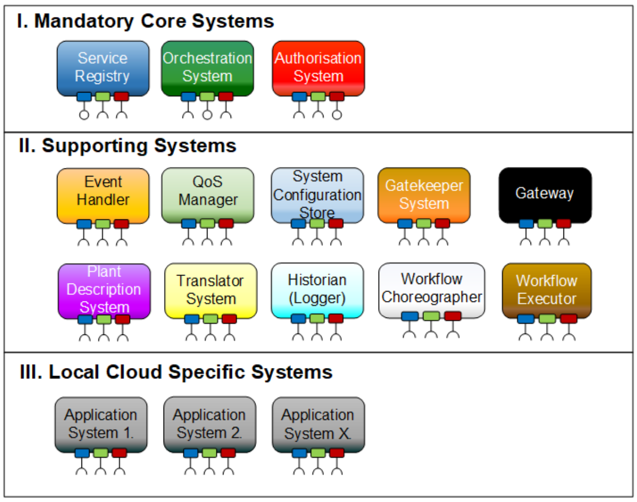


Fig. 7: System classification of Arrowhead for the given example [62]

C. Examples for classification of Arrowhead systems

In the following sections, an example is provided regarding the Arrowhead framework system-set: how to classify them to the presented Grades according to the proposed methodology. Table II summarizes the results of classification.

Grade I. - Orchestration System

This system provides the matchmaking within Arrowhead, basically participating in all sub-activities of the CCC models.

1) Information Sharing: An enabler through creating the connection between the service consumer and provider;
2) Resource Sharing: Enabling the consumer to connect the needed service resource and control the flow;
3) Decision Synchronization: Controlling the matchmaking process, often through information gathered from other Supporting Systems;
4) Joint Knowledge: Since the processes get to be transparent with coordination of the Orchestration System, the supply chain participants can get valuable knowledge about the current supply chain, its gaps, and how they can improve the supply chain processes;
5) Process Integration: With the usage of a common framework, the different processes of the supply chain participants will be integrated, which leads to everyone having the same understanding of the processes; therefore, the Orchestration System maintains the agreed processes;
6) Trust Enhancement: The service discovery is available between the remote and local cloud; therefore, the Orchestration System is also responsible for facilitating external communication over the Global Service Discovery mechanism;
7) Cooperation: Create the connection between the participants of the supply chain and control the cooperation.

Grade II. - Workflow Choreographer

This system manages workflows based on the production plan.

1) Information Sharing: It handles and provide useful, real-time information about the production;
2) Resource Sharing: The Workflow Choreographer controls the workflow in a distributed way. It instantiates the distributed Workflow Executors and allocates services to them based on the available resources;
3) Decision Synchronization: During the production, it manages distributed workflow execution and also brings smart decisions based on the unexpected events;
4) Joint Knowledge: After the workflow execution, useful knowledge can be gathered about the executed workflow, which opens the way for proactive maintenance and process optimization.

Grade III. - Temperature Sensor

The temperature sensor is located in a room of the factory.

1) Information Sharing: It provides information about its parameters such as temperature, general condition, location, among others.

D. Lifecycle and toolchain management in Arrowhead

As shown in Figure 7, the Arrowhead framework already has several systems – and the status of this ecosystem is constantly changing, where new systems will appear, old ones are restructured, or will be retired. Besides the expectation that all supply chain participants will work together using the

TABLE II: System classification of Arrowhead systems

CCC Sub activities	Orchestration System	Workflow Choreographer	Temperature Sensor
Information Sharing	x	x	x
Resource Sharing	x	x	
Joint knowledge	x	x	
Trust Enhancement	x		
Cooperation	x		
Decision Synchronization	x	x	
Process Integration	x		
Summary	**7**	**4**	**1**
Grade	**I.**	**II.**	**III.**

same processes and technological solutions, Arrowhead also allows participants to create their own Application Systems, which are then used locally or even be used by external actors. Accordingly, to make the development and integration dynamic, and even be accessible by external clouds, it is also necessary to determine the lifecycle of the systems and the tools used during the lifecycle. For this purpose, new lifecycle management [63], besides a new toolchain model [64] have been developed for Arrowhead. Although the trigger was to define models to the Arrowhead, these were universally designed to provide a good basis not only for the Arrowhead but for the SoS environments.

VI. CONCLUSION

The novelty of this paper is to introduce a newly designed development methodology for IoT-based SCM frameworks and platforms. First, the related literature was reviewed to understand the motivation behind, examining the main SCM modeling approaches and changes brought by Industry 4.0. Based on the approaches and changes, the challenges and the gaps were identified. As a result, IoT-based frameworks and platforms have been identified to be a solution to support DSCs. Accordingly, a specific IoT-SCM development methodology has been developed, which defines these platforms' characteristics by three main activities, namely the *Collaboration, Control* and *Combination*, and their related sub-activities. Also, the SOA has been proposed as the basic architecture of the methodology. Besides, a classification technique of the systems is introduced as well, covering the requirements of modularity and flexibility, among others. The presented methodology has been validated by an existing industrial IoT framework, the Arrowhead. The introduced development methodology presented by this paper can serve as useful guidance for IoT-SCM system developers. The model covers the needs of Industry 4.0 and DSC as well. Industry 4.0 is still under standardization; therefore, the requirements could change in the future, just like the expectations of DSCs. On the one hand, the IoT-SCM development methodology must adapt to the changing industrial trends. On the other hand, the relevance of the presented methodology needs further and continuous review and fine-tune.

REFERENCES

[1] M. Hermann, T. Pentek, and B. Otto, "Design Principles for Industrie 4.0 Scenarios," in *2016 49th Hawaii International Conference on System Sciences (HICSS)*. IEEE, January 2016. [Online]. Available: DOI: 10.1109/hicss.2016.488

[2] G. Büyüközkan and F. Göçer, "Digital Supply Chain: Literature review and a proposed framework for future research," *Computers in Industry*, vol. 97, pp. 157–177, May 2018. [Online]. Available: DOI: 10.1016/j.compind.2018.02.010

[3] M. C. Cooper, D. M. Lambert, and J. D. Pagh, "Supply Chain Management: More Than a New Name for Logistics," *The International Journal of Logistics Management*, vol. 8, no. 1, pp. 1–14, Jan 1997. [Online]. Available: DOI: 10.1108/09574099710805556

[4] D. M. Lambert and M. G. Enz, "Issues in Supply Chain Management: Progress and potential," *Industrial Marketing Management*, vol. 62, pp. 1–16, April 2017. [Online]. Available: DOI: 10.1016/j.indmarman.2016.12.002

[5] S. H. Huan, S. K. Sheoran, and G. Wang, "A review and analysis of supply chain operations reference (SCOR) model," *Supply Chain Management: An International Journal*, vol. 9, no. 1, pp. 23–29, February 2004. [Online]. Available: DOI: 10.1108/13598540410517557

[6] *APICS, SCOR - Supply Chain Operations Reference Model*, 2017.

[7] DIN Std., *Reference Architecture Model Industrie 4.0 (RAMI4.0)*, DIN SPEC 91345, April 2016.

[8] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall, January 2005.

[9] W. Kersten, T. Blecker, and C. M. Ringle, "Artificial Intelligence and Digital Transformation in Supply Chain Management: Innovative Approaches for Supply Chains," in *Proceedings of the Hamburg International Conference of Logistics (HICL) 27*. Berlin: Epubli GmbH, 2019.

[10] K. Korpela, J. Hallikas, and T. Dahlberg, "Digital Supply Chain Transformation toward Blockchain Integration," in *Proceedings of the 50th Hawaii International Conference on System Sciences (2017)*. Hawaii International Conference on System Sciences, 2017. [Online]. Available: DOI: 10.24251/hicss.2017.506

[11] E. Isikli, S. Yanik, E. Cevikcan, and A. Ustundag, "Project Portfolio Selection for the Digital Transformation Era," in *Springer Series in Advanced Manufacturing*. Springer International Publishing, September 2017, pp. 105–121. [Online]. Available: DOI: 10.1007/978-3-319-57870-5_6

[12] S. E. Chang, Y.-C. Chen, and M.-F. Lu, "Supply chain reengineering using blockchain technology: A case of smart contract based tracking process," *Technological Forecasting and Social Change*, vol. 144, pp. 1–11, July 2019. [Online]. Available: DOI: 10.1016/j.techfore.2019.03.015

[13] A. Rajeev, R. K. Pati, S. S. Padhi, and K. Govindan, "Evolution of sustainability in supply chain management: A literature review," *Journal of Cleaner Production*, vol. 162, pp. 299–314, September 2017. [Online]. Available: DOI: 10.1016/j.jclepro.2017.05.026

[14] T. Hines, Supply chain strategies: *Demand driven and customer focused*. Routledge, 2014.

[15] G. Milovanovic, S. Milovanovic, and G. Radisavljevic, "Globalization: The key challenge of modern supply chains," *Ekonomika*, vol. 63, no. 1, pp. 31–40, 2017. [Online]. Available: DOI: 10.5937/ekonomika1701031m

[16] D. Arunachalam, N. Kumar, and J. P. Kawalek, "Understanding big data analytics capabilities in supply chain management: Unravelling the issues, challenges and implications for practice," *Transportation Research Part E: Logistics and Transportation Review*, vol. 114, pp. 416–436, June 2018. [Online]. Available: DOI: 10.1016/j.tre.2017.04.001

[17] M. Christopher and J. Gattorna, "Supply chain cost management and value-based pricing," Industrial Marketing Management, vol. 34, no. 2, pp. 115–121, February 2005. [Online]. Available: DOI: 10.1016/j.indmarman.2004.07.016

[18] R. Angeles, "RFID Technologies: Supply-Chain Applications and Implementation Issues," Information Systems Management, vol. 22, no. 1, pp. 51–65, December 2005. [Online]. Available: DOI: 10.1201/1078/44912.22.1.20051201/85739.7

[19] S. E. Fawcett, S. L. Jones, and A. M. Fawcett, “Supply chain trust: The catalyst for collaborative innovation,” *Business Horizons*, vol. 55, no. 2, pp. 163–178, March 2012. [Online]. Available: DOI: 10.1016/j.bushor.2011.11.004

[20] M. S. Shahbaz, A. F. Chandio, M. Oad, A. Ahmed, R. Ullah, and “Stakeholders’ Management Approaches in Construction Supply Chain: A New Perspective of Stakeholder’s Theory,” *International Journal of Sustainable Construction Engineering and Technology*, vol. 9, no. 2, December 2018. [Online]. Available: DOI: 10.30880/ijscet.2018.09.02.002

[21] N. Pantlin, C. Wiseman, and M. Everett, “Supply chain arrangements: The ABC to GDPR compliance —A spotlight on emerging market practice in supplier contracts in light of the GDPR,” *Computer Law & Security Review*, vol. 34, no. 4, pp. 881–885, August 2018. [Online]. Available: DOI: 10.1016/j.clsr.2018.06.009

[22] European Parliament and Council of European Union, “Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation),” 2016.

[23] D. Kozma, P. Varga, and C. Hegedus, “Supply Chain Management and Logistics 4.0 - A Study on Arrowhead Framework Integration,” in *2019 8th International Conference on Industrial Technology and Management (ICITM)*. IEEE, March 2019. [Online]. Available: DOI: 10.1109/icitm.2019.8710670

[24] S. E. Sampson and C. M. Froehle, “Foundations and Implications of a Proposed Unified Services Theory,” *Production and Operations Management*, vol. 15, no. 2, pp. 329–343, January 2009. [Online]. Available: DOI: 10.1111/j.1937-5956.2006.tb00248.x

[25] ——, “Foundations and Implications of a Proposed Unified Services Theory,” *Production and Operations Management*, vol. 15, no. 2, pp. 329–343, January 2009. [Online]. Available: DOI: 10.1111/j.1937-5956.2006.tb00248.x

[26] T. P. Stank, S. B. Keller, and P. J. Daugherty, “Supply Chain Collaboration and Logistical Service Performance,” *Journal of Business Logistics*, vol. 22, no. 1, pp. 29–48, March 2001. [Online]. Available: DOI: 10.1002/j.2158-1592.2001.tb00158.x

[27] M. Barratt, “Understanding the meaning of collaboration in the supply chain,” *Supply Chain Management: An International Journal*, vol. 9, no. 1, pp. 30–42, Ferbruary 2004. [Online]. Available: DOI: 10.1108/13598540410517566

[28] L. Horvath, “Collaboration: the key to value creation in supply chain management,” *Supply Chain Management: An International Journal*, vol. 6, no. 5, pp. 205–207, December 2001. [Online]. Available: DOI: 10.1108/eum0000000006039

[29] Z. Cao, B. Huo, Y. Li, and X. Zhao, “The impact of organizational culture on supply chain integration: a contingency and configuration approach,” *Supply Chain Management: An International Journal*, vol. 20, no. 1, pp. 24–41, January 2015. [Online]. Available: DOI: 10.1108/scm-11-2013-0426

[30] D. J. Bowersox, D. J. Closs, and T. P. Stank, *21st century logistics: making supply chain integration a reality*. Council of Supply Chain Management Professionals, 1999.

[31] D. Prajogo and J. Olhager, “Supply chain integration and performance: The effects of long-term relationships, information technology and sharing, and logistics integration,” *International Journal of Production Economics*, vol. 135, no. 1, pp. 514–522, January 2012. [Online]. Available: DOI: 10.1016/j.ijpe.2011.09.001

[32] T. E. Vollmann, *Manufacturing planning and control for supply chain management*. McGraw-Hill Education, 2005.

[33] H. Sarimveis, P. Patrinos, C. D. Tarantilis, and C. T. Kiranoudis, “Dynamic modeling and control of supply chain systems: A review,” *Computers & Operations Research*, vol. 35, no. 11, pp. 3530–3561, November 2008. [Online]. Available: DOI: 10.1016/j.cor.2007.01.017

[34] L. S. Dias and M. G. Ierapetritou, “From process control to supply chain management: An overview of integrated decision making strategies,” *Computers & Chemical Engineering*, vol. 106, pp. 826–835, November 2017. [Online]. Available: DOI: 10.1016/j.compchemeng.2017.02.006

[35] D. Kozma, P. Varga, and G. Soos, “Supporting Digital Production, Product Lifecycle and Supply Chain Management in Industry 4.0 by the Arrowhead Framework – a Survey,” in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*. IEEE, July 2019. [Online]. Available: DOI: 10.1109Findin41052.2019.8972216

[36] R. Handfield, E. Nichols, and E. Nichols, *Introduction to Supply Chain Management. Prentice Hall*, 1999.

[37] J. T. Mentzer, W. DeWitt, J. S. Keebler, S. Min, N. W. Nix, C. D. Smith, and Z. G. Zacharia, “Defining Supply Chain Management,” *Journal of Business Logistics*, vol. 22, no. 2, pp. 1–25, 2001. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/j.2158-1592.2001.tb00001.x

[38] M. Christopher, *Logistics & supply chain management*. Pearson UK, 2016.

[39] T. M. Simatupang and R. Sridharan, “The collaboration index: a measure for supply chain collaboration,” *International Journal of Physical Distribution & Logistics Management*, vol. 35, no. 1, pp. 44–62, January 2005. [Online]. Available: DOI: 10.1108/09600030510577421

[40] K. Goffin, F. Lemke, and M. Szwejczewski, “An exploratory study of ‘close’ supplier-manufacturer relationships,” *Journal of Operations Management*, vol. 24, no. 2, pp. 189–209, July 2005. [Online]. Available: DOI: 10.1016/j.jom.2005.05.003

[41] B. Sahay, “Understanding trust in supply chain relationships,” *Industrial Management & Data Systems*, vol. 103, no. 8, pp. 553–563, November 2003. [Online]. Available: DOI: 10.1108/02635570310497602

[42] M. Cao and Q. Zhang, *Supply Chain Collaboration: Roles of Interorganizational Systems, Trust, and Collaborative Culture*. Springer Science & Business Media, 2012.

[43] Malhotra, Gosain, and Sawy, “Absorptive Capacity Configurations in Supply Chains: Gearing for Partner-Enabled Market Knowledge Creation,” *MIS Quarterly*, vol. 29, no. 1, p. 145, 2005. [Online]. Available: DOI: 10.2307/25148671

[44] Arshinder, A. Kanda, and S. Deshmukh, “Supply chain coordination: Perspectives, empirical studies and research directions,” *International Journal of Production Economics*, vol. 115, no. 2, pp. 316–335, October 2008. [Online]. Available: DOI: 10.1016/j.ijpe.2008.05.011

[45] F. Chen, “Information Sharing and Supply Chain Coordination,” in *Supply Chain Management: Design, Coordination and Operation*. Elsevier, 2003, pp. 341–421. [Online]. Available: DOI: 10.1016/s0927-0507/803/911007-9

[46] X. Li and Q. Wang, “Coordination mechanisms of supply chain systems,” *European Journal of Operational Research*, vol. 179, no. 1, pp. 1–16, May 2007. [Online]. Available: DOI: 10.1016/j.ejor.2006.06.023

[47] B. Christiansen, *Handbook of Research on Global Supply Chain Management*. IGI Global, 2015.

[48] C. Mathuramaytha, “Supply Chain Collaboration – What’s an outcome?: A Theoretical Model,” in *2011 International Conference on Financial Management and Economics IPEDR* vol.11, Singapore, 2011.

[49] G. Disterer, *ISO/IEC 27000, 27001 and 27002 for Information Security Management*, 2013. [Online]. Available: DOI: 10.4236/jis.2013.42011

[50] NIST and E. Aroms, *NIST SP 800-100 Information Security Handbook: A Guide for Managers*. Scotts Valley, CA: CreateSpace, 2012.

[51] AICPA, *SOC 2^{TM} - SOC for Service Organizations: Trust Services Criteria*, 2018.

[52] J. W. Lainhart, “COBITTM: A methodology for managing and controlling information and information technology risks and vulnerabilities,” *Journal of Information Systems*, vol. 14, no. s-1, pp. 21–25, January 2000. [Online]. Available: DOI: 10.2308/jis.2000.14.s-1.21

[53] L. Roalter, A. Moller, S. Diewald, and M. Kranz, “Developing Intelligent Environments: A Development Tool Chain for Creation, Testing and Simulation of Smart and Intelligent Environments,” in *2011 Seventh International Conference on Intelligent Environments*. IEEE, July 2011. [Online]. Available: DOI: 10.1109/ie.2011.43

[54] D. Kozma, P. Varga, and F. Larrinaga, “Dynamic Multilevel Workflow Management Concept for Industrial IoT Systems,” *IEEE Transactions on Automation Science and Engineering*, pp. 1–13, 2020. [Online]. Available: DOI: 10.1109/tase.2020.3004313

[55] G. Urgese, P. Azzoni, J. van Deventer, J. Delsing, and E. Macii, "An Engineering Process model for managing a digitalised life-cycle of products in the Industry 4.0," in *NOMS 2020 - 2020 IEEE/IFIP Network Operations and Management Symposium*. IEEE, April 2020. [Online]. Available: DOI: 10.1109/noms47738.2020.9110365

[56] J. Delsing, Ed., *IoT Automation*. CRC Press, February 2017. [Online]. Available: DOI: 10.1201/9781315367897

[57] C. Hegedus, P. Varga, and A. Franko, "Secure and Trusted Intercloud Communications in the Arrowhead Framework," in *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*. IEEE, May 2018. [Online]. Available: DOI: 10.1109/icphys.2018.8390802

[58] M. Albano, L. Lino Ferreira, and J. Sousa, "Event Handler System: Publish/Subscribe communication for the Arrowhead world," in *12th IEEE World Conference on Factory Communication Systems (WFCS)*, 2016.

[59] O. Carlsson, D. Vera, J. Delsing, B. Ahmad, and R. Harrison, "Plant descriptions for engineering tool interoperability," in *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*. IEEE, July 2016. [Online]. Available: DOI: 10.1109/indin.2016.7819255

[60] H. Derhamy, J. Eliasson, J. Delsing, P. P. Pereira, and P. Varga, "Translation error handling for multi-protocol SOA systems," in *2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA)*. IEEE, September 2015. [Online]. Available: DOI: 10.1109/etfa.2015.7301473

[61] P. Varga, D. Kozma, and C. Hegedus, "Data-Driven Workflow Execution in Service Oriented IoT Architectures," in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, September 2018. [Online]. Available: DOI: 10.1109/etfa.2018.8502665

[62] D. Kozma, P. Varga, and F. Larrinaga, "Data-driven Workflow Management by utilising BPMN and CPN in IIoT Systems with the Arrowhead Framework," in *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, September 2019. [Online]. Available: DOI: 10.1109/etfa.2019.8869501

[63] ——, "System of Systems Lifecycle Management - A new concept based on process engineering methodologies," *Journal of Intelligent Manufacturing*, 2021, Accepted.

[64] G. Kulcsar, M. S. Tatara, and F. Montori, "Toolchain Modeling: Comprehensive Engineering Plans for Industry 4.0," in *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, October 2020. [Online]. Available: DOI: 10.1109/iecon43393.2020.9254855

Dániel Kozma received his MSc degree in Electrical Engineering at Budapest University of Technology and Economics (BME), Hungary, in 2015. He is an experienced database software engineer with a demonstrated history of working in telecommunications. He is currently working as an Information Security Officer. In parallel, he is also a researcher as a Ph.D. student at BME within the Productive 4.0 project. His research focuses on the different areas of Industry 4.0, such as automated production, supply chain- and lifecycle management, furthermore cyber- , and information security.

Pal Varga is currently an Associate Professor at the Budapest University of Technology and Economics, and Director at AITIA International Inc.

His main research interests include communication systems, network performance measurements, root cause analysis, fault localisation, traffic classification, end-to-end QoS and SLA issues – for which he is keen to apply hardware acceleration and artificial intelligence techniques as well. Recently he has been actively engaged with research related to Cyber-Physical Systems and Industrial Internet of Things.

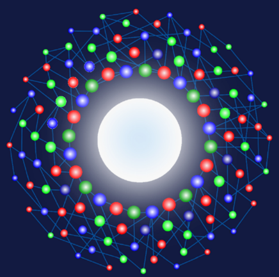
He has been involved in various industrial as well as European research and development projects in these topics.

25th International Conference on Optical Network Design and Modelling

June 28 – July 1, 2021 Gothenburgh Sweden

Call for Papers

https://ondm2021.chalmers.se/



ONDM 2021

Following the tradition of previous editions, ONDM 2021 addresses cutting-edge research in established areas of optical networking and their adoption in support of a wide variety of new services and applications. This includes the most recent trends such as 5G and beyond; data-centre networking; Internet of things; cloud/edge computing; content delivery; big data, data analytics, network telemetry and real-time monitoring; autonomic networking; artificial intelligence / machine learning assisted networks; visible light networks; and quantum secured networks.

Such trends drive the need for increased capacity, efficiency, flexibility, and adaptability in the functions that the network can perform. In turn, these needs can be achieved by developing new optical network architectures, based on different optical network disaggregation models, exploiting and integrating novel multidimensional photonic technology solutions as well as by adopting open and highly programmable hardware and software platforms such as software defined networking (SDN), and network function virtualization (NFV), to allow supporting new business models and opportunities. The scope of the conference includes but is not limited to the following topics:

- Advances in optical network modelling and optimization
- Routing and spectrum assignment in fixed and flex-grid optical networks
- Optical network availability, resilience, survivability, security and privacy
- Multi-layer (e.g., IP over optical) networking
- Optical networks exploiting photonic integrated circuits
- Visible light communications and networks
- Optical and wireless network convergence, including radio-over-fibre access networks
- Optical networks in 5G and beyond: backhaul, midhaul and fronthaul networking
- Novel and multidimensional optical network architectures; multi-mode/few-modes optical networks
- Novel optical node designs including disaggregation and open optical line systems
- Optical networks in support of intra-/inter-data centre connectivity and cloud/edge computing
- Optical network control, management and orchestration including SDN and NFV solutions
- Slicing, service chaining, virtualization and multi-tenancy techniques for optical networks
- Optical networking supporting low latency and high bandwidth network function virtualization
- Orchestration and control of IT and network resources in optical data-centre networks
- Artificial Intelligence and data analytics techniques for optical networks
- Novel network telemetry and real-time monitoring technologies for optical networks
- Energy efficiency in optical networks
- Transporting Power over Fibre
- The hollow fibre revolution in network design
- Optical networking in support of vertical industries
- Field trials and interoperability demonstrations of optical networks
- Techno-economic studies of optical network

ONDM2021 Chairs

General Chairs

Paolo Monti, Chalmers University of Technology, Sweden
Marija Furdek, Chalmers University of Technology, Sweden

TPC Chairs

Carmen Mas Machuca, Technical University of Munich (TUM), Germany
Jarosław Turkiewicz, Warsaw University of Technology (WUT), Poland
David Larrabeiti, University Carlos III of Madrid (UC3M), Spain

ONDM Steering Committee

Pablo Pavón Mariño, Universidad Politécnica de Cartagena (UPCT), Spain
Tibor Cinkler, Budapest University of Technology and Economics (BME), Hungary
Marco Ruffini, Trinity College Dublin, Ireland
Anna Tzanakaki, National and Kapodistrian University of Athens, Greece
Raül Muñoz, CTTC, Barcelona, Spain
Paolo Monti, Chalmers University of Technology, Sweden

Important Dates

Submission Deadline: March 12, 2021 • Acceptance Notification: May 14, 2021 • Camera Ready Submission: June 4, 2021

Speech Synthesis Workshop (SSW)

At an international conference on speech processing, a speech scientist once held up a tube of toothpaste (whose brand was "Signal") and, squeezing it in front of the audience, coined the phrase "This is speech synthesis; speech recognition is the art of pushing the toothpaste back into the tube."

One could turn this very simplistic view the other way round: users are generally much more tolerant of speech recognition errors than they are willing to listen to unnatural speech. There is magic in a speech recognizer that transcribes continuous radio speech into text with a word accuracy as low as 50%; in contrast, even a perfectly intelligible speech synthesizer is only moderately tolerated by users if it delivers nothing more than "robot voices". Delivering both intelligibility and naturalness has been the holy grail of speech synthesis research for the past 30 years. More recently, expressivity has been added as a major objective of speech synthesis.

Add to this the engineering costs (computational cost, memory cost, design cost for making another synthetic voice or another language) which have to be taken into account, and you'll start to have an idea of the challenges underlying text-to-speech synthesis.

Major challenges call for major meetings: the Speech Synthesis Workshops (SSWs), which are held every three years under the auspices of ISCA's SynSIG. In 2019 it was decided to have an SSW every two years, since the technology is advancing faster these days. SSWs provide a unique occasion for people in the speech synthesis area to meet each other. They contribute to establishing a feeling that we are all participating in a joint effort towards intelligible, natural, and expressive synthetic speech.

Workshop Topics

Papers in all areas of speech synthesis technology are encouraged to be submitted, including but not limited to:

- Grapheme-to-phoneme conversion for synthesis
- Text processing for speech synthesis (text normalization, syntactic and semantic analysis, intent detection)
- Segmental-level and/or concatenative synthesis
- Signal processing/statistical model for synthesis
- Speech synthesis paradigms and methods; articulatory synthesis, articulation-to-speech synthesis, parametric synthesis etc.
- Prosody modeling, transfer and generation
- Expression, emotion and personality generation
- Voice conversion and modification, morphing (parallel and non-parallel)
- Concept-to-speech conversion speech synthesis in dialog systems
- Avatars and talking faces
- Cross-lingual and multilingual aspects for synthesis (e.g. automatic language switching)
- Applications of synthesis technologies to communication disorders
- TTS for embedded devices and computational issues
- Tools and data for speech synthesis
- Quality assessment/evaluation metrics in synthesis
- End-to-end text-to-speech synthesis
- Direct speech waveform modelling and generation
- Neural vocoding for speech synthesis
- Speech synthesis using non-ideal data ('found', user-contributed, etc.)
- Natural language generation for speech synthesis
- Special topic: Speech uniqueness and deep learning (generating diverse and natural speech)

Call for Papers

The workshop program will consist of a single track with invited talks, oral and poster presentations. Prospective authors are invited to submit original, full-length, 4-6 page papers, including figures and references. Papers can be submitted via the Easychair website in April-May, 2021.

Please follow the INTERSPEECH 2021 guidelines and templates when preparing your paper. Papers have to follow these guidelines except that we allow up to 6 pages, including figures and references.

Papers will be published in the ISCA online archive.

Call for Demos

We are planning to have a demo session to showcase new developments in speech synthesis. If you have some demonstrations of your work that does not really fit in a regular oral or poster presentation, please let us know.

Keynotes

Thomas Drugman *Amazon, Germany*
Expressive Neural TTS
István Winkler *Research Centre for Natural Sciences, Hungary*
Early Development of Infantile Communication by Sound
Lior Wolf *Facebook AI Research and Tel Aviv University, Israel*
Deep Audio Conversion Technologies and Their Applications in Speech, Singing, and Music

Venue

VITAL HOTEL NAUTIS **** wellness and conference hotel in Gárdony, the capital of Lake Velence directly on the lakeshore, next to the port and the beach.

Organizing Committee

Géza Németh *Budapest University of Technology and Economics, Department of Telecommunications and Media Informatics (BME TMIT), Chairman*
Junichi Yamagishi *National Institute of Informatics Japan, University of Edinburgh, UK*
Sébastien Le Maguer *ADAPT Centre/TCD, Ireland*
Esther Klabbers *Readspeaker, Netherlands*
Mátyás Bartalis *BME TMIT, Hungary*
Tamás Gábor Csapó *BME TMIT, Hungary*
Bálint Gyires-Tóth *BME TMIT, Hungary*
Gábor Olaszy *BME TMIT, Hungary*
Csaba Zainkó *BME TMIT, Hungary*

Important Dates

April-May, 2021	Paper submission
May, 2021	Registration opens
June, 2021	Notification of acceptance
26-28 August, 2021	Workshop

HTE - HÍRKÖZLÉSI ÉS INFORMATIKAI TUDOMÁNYOS EGYESÜLET - SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS
H-1051 Budapest, Bajcsy-Zsilinszky út 12. Hungary • Phone: +36 1 353 1027 • www.hte.hu • e-mail: info@hte.hu

17th International Conference on Network and Service Management

Smart Management for Future Networks and Services

25-29 October 2021, Izmir, Turkey

CALL FOR PAPERS

The 17th International Conference on Network and Service Management (CNSM) is inviting authors to submit original contributions in network and service management research. CNSM 2021 is a selective single-track conference, covering all aspects of the management of networks and services, pervasive systems, enterprises, and cloud computing environments. The core track will be accompanied by a series of workshops and poster sessions.

Papers accepted and presented at CNSM 2021 will be published as open access on the conference website and will be submitted to IEEE Xplore Digital Library. Authors of selected papers, accepted for publication in the CNSM 2021 proceedings, will be invited to submit an extended version of their papers to the IEEE Transactions on Network and Service Management journal.

Topics of Interest (but not limited to)

Network Management
- Software-defined networks
- Virtual networks
- Overlay networks
- Wireless and cellular networks
- Sensor networks
- Internet of Things
- Information-centric networks
- Enterprise and campus networks
- Data center networks
- Optical networks
- Home networks
- Access networks
- Smart cities and Smart grids

Management Technologies
- Network function virtualization
- Software-defined networking
- Orchestration
- Network slicing
- Service function chaining
- Cloud computing and cloud storage
- Edge/Fog computing
- Middleware
- Data models and semantic models
- Operations support systems and business support systems

Service Management
- Cloud computing services
- Content delivery services
- Multimedia services
- Internet connectivity and Internet access
- Internet of Things services
- Security services
- Context-aware services
- Information technology services

Methods
- Mathematical optimization
- Control theory
- Probability theory, stochastic processes, queuing theory
- Machine learning, deep learning, artificial intelligence
- Federated learning
- Evolutionary computing
- Economic theory and game theory
- Mathematical logic and reasoning
- Data mining and (big) data analysis
- Monitoring and measurements
- Computer simulation experiments
- Prototype implementation and testbed experimentation
- Field trials

Management Paradigms
- Centralized management
- Distributed management
- Hierarchical management
- Federated management
- Autonomic and cognitive management
- Policy-based management
- Intent-Based management
- Pro-active management
- Energy-aware management
- Quality of experience-centric management

Business Management
- Economic/financial aspects
- Multi-stakeholder aspects
- Service level agreements
- Lifecycle aspects
- Process & workflow aspects
- Privacy aspects

Functional Areas
- Fault management
- Configuration management
- Performance management
- Security management
- Accounting Management

Authors are invited to submit original contributions that have not been published or submitted for publication elsewhere. Papers should be prepared using the IEEE 2-column conference style and are limited to 9 pages including references (full papers) or 5 pages including references (short papers). They have to be submitted electronically in PDF format through EDAS at https://www.edas.info/newPaper.php?c=28015

Papers exceeding page limits, multiple submissions, and self-plagiarized papers will be rejected without further review. All other papers will get a thorough single-blind review process, followed by a rebuttal phase.

For more information please check http://www.cnsm-conf.org/2021/

Important Dates

Paper Submission:	***8 May 2021***
Acceptance Notification:	***15 July 2021***
Camera Ready due:	***1 Aug 2021***

Technical Program Co-Chairs:
Müge Sayıt, Ege University, Turkey
Stuart Clayman, University College London, UK

General Co-Chairs:
Prosper Chemouil, Cnam, France
Mehmet Ulema, Manhattan College, USA

http://www.cnsm-conf.org/2021/

Guidelines for our Authors

Format of the manuscripts

Original manuscripts and final versions of papers should be submitted in IEEE format according to the formatting instructions available on
https://journals.ieeeauthorcenter.ieee.org/
Then click: "IEEE Author Tools for Journals"
- "Article Templates"
- "Templates for Transactions".

Length of the manuscripts

The length of papers in the aforementioned format should be 6-8 journal pages.
Wherever appropriate, include 1-2 figures or tables per journal page.

Paper structure

Papers should follow the standard structure, consisting of *Introduction* (the part of paper numbered by “1”), and *Conclusion* (the last numbered part) and several *Sections* in between.
The Introduction should introduce the topic, tell why the subject of the paper is important, summarize the state of the art with references to existing works and underline the main innovative results of the paper. The Introduction should conclude with outlining the structure of the paper.

Accompanying parts

Papers should be accompanied by an *Abstract* and a few *index terms (Keywords)*. For the final version of accepted papers, please send the short cvs and *photos* of the authors as well.

Authors

In the title of the paper, authors are listed in the order given in the submitted manuscript. Their full affiliations and e-mail addresses will be given in a footnote on the first page as shown in the template. No degrees or other titles of the authors are given. Memberships of IEEE, HTE and other professional societies will be indicated so please supply this information. When submitting the manuscript, one of the authors should be indicated as corresponding author providing his/her postal address, fax number and telephone number for eventual correspondence and communication with the Editorial Board.

References

References should be listed at the end of the paper in the IEEE format, see below:

a) Last name of author or authors and first name or initials, or name of organization
b) Title of article in quotation marks
c) Title of periodical in full and set in italics
d) Volume, number, and, if available, part
e) First and last pages of article
f) Date of issue
g) Document Object Identifier (DOI)

[11] Boggs, S.A. and Fujimoto, N., “Techniques and instrumentation for measurement of transients in gas-insulated switchgear,” IEEE Transactions on Electrical Installation, vol. ET-19, no. 2, pp.87–92, April 1984. DOI: 10.1109/TEI.1984.298778

Format of a book reference:

[26] Peck, R.B., Hanson, W.E., and Thornburn, T.H., Foundation Engineering, 2nd ed. New York: McGraw-Hill, 1972, pp.230–292.

All references should be referred by the corresponding numbers in the text.

Figures

Figures should be black-and-white, clear, and drawn by the authors. Do not use figures or pictures downloaded from the Internet. Figures and pictures should be submitted also as separate files. Captions are obligatory. Within the text, references should be made by figure numbers, e.g. “see Fig. 2.”
When using figures from other printed materials, exact references and note on copyright should be included. Obtaining the copyright is the responsibility of authors.

Contact address

Authors are requested to submit their papers electronically via the following portal address:

https://www.ojs.hte.hu/index.php/infocommunications_journal/about/submissions

If you have any question about the journal or the submission process, please do not hesitate to contact us via e-mail:

Editor-in-Chief: Pál Varga – pvarga@tmit.bme.hu

Associate Editor-in-Chief:
Rolland Vida – vida@tmit.bme.hu
László Bacsárdi – bacsardi@hit.bme.hu

IEEE GLOBECOM® 2021

IEEE ComSoc™ IEEE Communications Society

IEEE Global Communications Conference

7-11 December 2021 • Madrid, Spain

Connecting Cultures around the Globe

CALL FOR PAPERS AND PROPOSALS

The 2021 IEEE Global Communications Conference (GLOBECOM) will be held in Madrid, Spain, from 7 -11 December 2021. Themed *“Connecting Cultures around the Globe,”* this flagship conference of the IEEE Communications Society will feature a comprehensive high-quality technical program including 13 symposia and a variety of tutorials and workshops. IEEE GLOBECOM 2021 will also include an attractive Industry program aimed at practitioners, with keynotes and panels from prominent research, industry and government leaders, business and industry panels, and vendor exhibits.

TECHNICAL SYMPOSIA

- Cognitive Radio and AI-Enabled Networks
- Communication and Information System Security
- Communication QoS, Reliability and Modeling
- Communication Software and Multimedia
- Communication Theory
- Green Communication Systems and Networks
- IoT and Sensor Networks
- Mobile & Wireless Networks
- Next-Generation Networking and Internet
- Optical Networks & Systems
- Signal Processing for Communications
- Wireless Communications
- Selected Areas in Communications
  - *Aerial Communications*
  - *Big Data*
  - *Cloud Computing, Networking and Storage*
  - *E-Health*
  - *Full-Duplex Communications*
  - *Machine Learning for Communications*
  - *Molecular, Biological and Multi-Scale Communications*
  - *Quantum Communications & Computing*
  - *Satellite and Space Communications*
  - *Smart Grid Communications*
  - *Social Networks*

INDUSTRY FORUMS AND EXHIBITION PROGRAM

Proposals are sought for forums, panels, demos, seminars and presentations specifically related to issues facing the broader communications and networking industries.

TUTORIALS

Proposals are sought for half- or full-day tutorials in all communication and networking topics.

WORKSHOPS

Proposals are sought for half- or full-day workshops in all communication and networking topics.

IMPORTANT DATES

Paper Submission 15 April 2021 (23:59)	**Tutorial Proposals** 15 April 2021
Acceptance Notification 25 July 2021	**Workshop Proposals** 15 March 2021
Camera-Ready 1 September 2021	**Technical Panel Proposals** 1 June 2021

Full details of submission procedures are available at globecom2021.ieee-globecom.org

ORGANIZING COMMITTEE

General Chairs
Juan Manuel Corchado, University of Salamanca, Spain
Ana García Armada, Universidad Carlos III de Madrid, Spain

Executive Chair
Joel Rodrigues, INATEL, Brazil

Executive Vice-Chair
Víctor Gil, Universidad Carlos III de Madrid, Spain

Technical Program Chair
Sennur Ulukus, University of Maryland, USA

Technical Program Vice Chairs
Rui Dinis, FCT-UNL, Portugal
Santiago Mazuelas, BCAM, Spain

Tutorials Chairs
Octavia Dobre, Memorial University, Canada
Pascal Lorenz, University of Haute-Alsace, France

Workshops Chairs
Mohsen Guizani, University of Idaho, USA
Guangjie Han, Hohai University, China

Publications Chairs
Sofiène Affes, INRS, Canada
Mutlu Koca, Boğaziçi University, Turkey

Industry Forums & Exhibition Chairs
Enrique Díaz-Plaza, IBM, Spain
Jaime Lloret, Universitat Politècnica de València, Spain

Operations Chair
Javier Prieto, University of Salamanca, Spain

globecom2021.ieee-globecom.org

SCIENTIFIC ASSOCIATION FOR INFOCOMMUNICATIONS

hte

FOUNDED IN 1949

SCIENTIFIC
ASSOCIATION FOR
INFOCOMMUNICATIONS
HUNGARY

Who we are

Founded in 1949, the Scientific Association for Infocommunications (formerly known as Scientific Society for Telecommunications) is a voluntary and autonomous professional society of engineers and economists, researchers and businessmen, managers and educational, regulatory and other professionals working in the fields of telecommunications, broadcasting, electronics, information and media technologies in Hungary.

Besides its 1000 individual members, the Scientific Association for Infocommunications (in Hungarian: Hírközlési és Informatikai Tudományos Egyesület, HTE) has more than 60 corporate members as well. Among them there are large companies and small-and-medium enterprises with industrial, trade, service-providing, research and development activities, as well as educational institutions and research centers.

HTE is a Sister Society of the Institute of Electrical and Electronics Engineers, Inc. (IEEE) and the IEEE Communications Society.

What we do

HTE has a broad range of activities that aim to promote the convergence of information and communication technologies and the deployment of synergic applications and services, to broaden the knowledge and skills of our members, to facilitate the exchange of ideas and experiences, as well as to integrate and harmonize the professional opinions and standpoints derived from various group interests and market dynamics.

To achieve these goals, we…

- contribute to the analysis of technical, economic, and social questions related to our field of competence, and forward the synthesized opinion of our experts to scientific, legislative, industrial and educational organizations and institutions;
- follow the national and international trends and results related to our field of competence, foster the professional and business relations between foreign and Hungarian companies and institutes;
- organize an extensive range of lectures, seminars, debates, conferences, exhibitions, company presentations, and club events in order to transfer and deploy scientific, technical and economic knowledge and skills;
- promote professional secondary and higher education and take active part in the development of professional education, teaching and training;
- establish and maintain relations with other domestic and foreign fellow associations, IEEE sister societies;
- award prizes for outstanding scientific, educational, managerial, commercial and/or societal activities and achievements in the fields of infocommunication.

Contact information

President: **FERENC VÁGUJHELYI** • *elnok@hte.hu*
Secretary-General: **ISTVÁN MARADI** • *istvan.maradi@gmail.com*
Operations Director: **PÉTER NAGY** • *nagy.peter@hte.hu*
International Affairs: **ROLLAND VIDA, PhD** • *vida@tmit.bme.hu*

Address: H-1051 Budapest, Bajcsy-Zsilinszky str. 12, HUNGARY, Room: 502
Phone: +36 1 353 1027
E-mail: *info@hte.hu*, Web: *www.hte.hu*