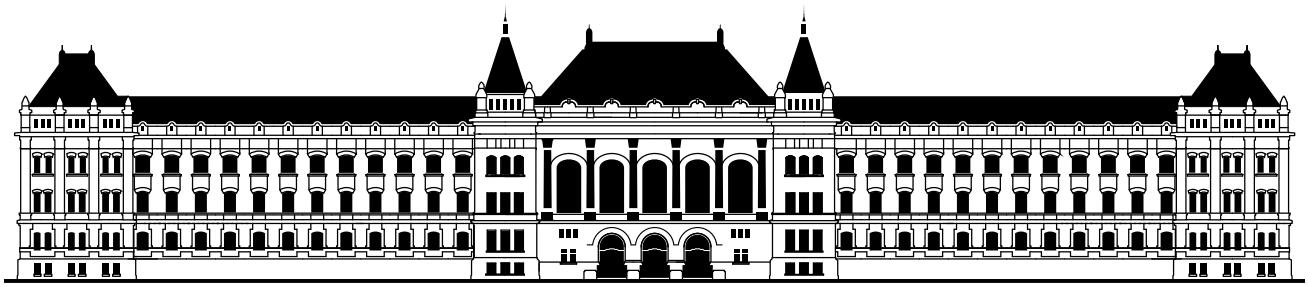
PROCEEDINGS OF THE 29TH MINISYMPOSIUM

OF THE

DEPARTMENT OF MEASUREMENT AND INFORMATION SYSTEMS
BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS

(MINISY@DMIS 2022)

FEBRUARY 7–8, 2022
BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS



BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS
DEPARTMENT OF MEASUREMENT AND INFORMATION SYSTEMS

ISBN 978-963-421-872-2

Head of Department:
Tamás Dabóczi

General Chair:
Balázs Renczes

Scientific Chairs:
Ákos Jobbágy
András Pataricza
Tadeusz Dobrowiecki

Local Chairs:
Márton Elekes
Kristóf Horváth
András Palkó

Homepage of the Conference:
`http://minisy.mit.bme.hu/`

Sponsored by:
Schnell László Foundation

FOREWORD

On behalf of the Organizing Committee, I would like to greet you at the 29th Minisymposium of the Department of Measurement and Information Systems at the Budapest University of Technology and Economics.

After the complete lockdown at the university due to the pandemia last year, we are glad to hold the Symposium in person again. It is an honor to experience that besides Ph.D. and master students, we can also welcome several researchers from our widespread international connections.

As we have seen in the previous years, the covered topics are fairly diversified: among others, we will have presentations from the area of digital signal processing, bio-medicine and bio-informatics, security, artificial intelligence, and cyber-physical systems.

Following the practice of the last couple of years, we issue the proceedings only in electronic format. We believe that the advantages of this format make it unnecessary in the future to publish a printed edition.

I hope that the following two days will be fruitful in the sense that you will be able to exchange ideas, grasp inspiration from the research approaches of one another, or even discover areas where further international co-operations can be established.

Budapest, February 7, 2022

Balázs Renczes
General Chair

PAPERS OF THE MINISYMPOSIUM

An Initial Performance Analysis of Graph Predicate Evaluation over Partial Models

Attila Ficsor, Oszkár Semeráth
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: attila.ficsor@edu.bme.hu, semerath@mit.bme.hu

***Abstract*—Graph-based modeling tools are widely used during the design, analysis and verification of complex critical systems. Those tools enables the automation of several design steps (e.g., by model transformation), and the early analysis of system designs (e.g. by test generation). The evaluation of complex graph predicates (or graph pattern matching) is a core technique in modeling and model transformation, and essential in scalable graph generation. This motivated the integration of industrial graph pattern matching tools directly to advanced data structures used in model checking and logic reasoning algorithms.**

In this paper we provide a report of a preliminary performance benchmark combining the incremental graph pattern matching algorithm of the Viatra framework with hash tries used for state space exploration on partial models.

***Index Terms*—predicate evaluator, graph generation**

I. Model generation and predicate evaluation

During the design and testing of critical systems, modeling tools are widely used, enabling the automation of several development and testing steps with graph-based models. Graph-based models are the primary development artifacts in those modeling environments on which advanced modeling frameworks are operating. To test those modeling applications, we need a diverse set of well-formed models as test input.

However, the synthesis of valid well-formed models is a challenging task. A common feature of scalable model synthesis algorithms is the continuous evaluation of graph predicates during an exploration process:

- The VIATRA Solver model generation approach [1] combines the incremental graph pattern matching [2] with rule-based design exploration framework [3].
- The SDG framework [4] combines standard OCL-based tooling [5] with genetic algorithms.

In this paper we compare the performance of a new prototype predicate evaluation technique using hash tries [6] for efficiently storing multiple versions of a graph models, and the incremental graph pattern matching algorithm [2]. We present the measurements on a case study, where we generate diverse and realistic scenarios for testing machine learning components used in advanced driver-assistance systems.

II. Challenges of modeling technologies

Until now, the only documented way to create patterns for VIATRA was in VIATRA Query Language [2]. To use this, we needed to work our way through a long list of steps setting up the integrated development environment (IDE). This included installing Eclipse with Eclipse Modeling Framework (EMF) and VIATRA, then creating a modeling project, where we could create a metamodel. From this we had to generate model code and editor code, which we had to use to start a Runtime Eclipse. In this instance of Eclipse we could create a Query Project, in it a VQL file, and in this file, we could write our pattern in VQL language. When we saved this file, some Java classes were generated, that we could use in our code.

This method used EMF objects to store data (i.e. the model), and there was limited options to use other data structures. In the existing code base in the VIATRA framework, there are two ways to create (partial) models used as the starting point of the generation. One is to create an EMF model either using the graphical user interface (GUI) editor, or using Java programs. We can then load this model, and VIATRA builds its own internal data structure from the model. The other way we can create a partial model is using a tabular method, where we can create a table which we can use to write our data into. Then based on this table, VIATRA creates its own internal data structure, similar to the previous method. Unfortunately, this is implemented for data types provided by EMF.

There are several challenges resulting from this:

- Setting up the IDE is not user-friendly, it has complicated software requirements and necessary settings, that are difficult to find.
- Portability is limited, since one version has Eclipse dependency, while the other version without this dependency is unstable.
- Originally, the pattern matching operates on data structures provided by EMF, which imposes performance limitations (e.g., scalability issues with ELists and inefficient state space exploration using EMF transactions).

To answer those challenges, we chose to integrate high performance data structures [6] to the core pattern matching mechanism of the VIATRA query framework.

- We provide a simple grammar to formulate type systems, the predicates (i.e., the queries) and instance models [7].
- The framework can be used without custom editors.
- Finally, [6] promises efficient and scalable data structures optimized for exploring huge search spaces necessitated by explicit model checking algorithms.

DOI 10.3311/MINISY2022-001

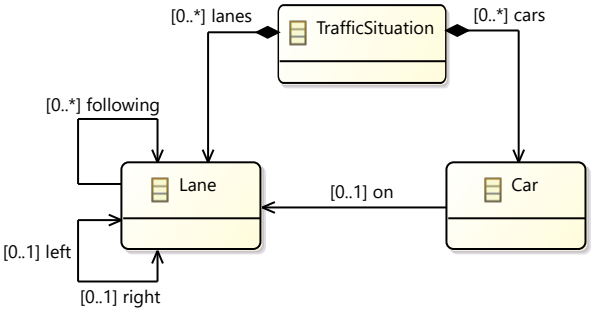


Fig. 1: EMF metamodel used in the measurements

In this paper, our main goal is to provide an initial performance comparison between the newly developed data structure and VIATRA. In the following, we present the domain we selected to execute our performance comparison. For this example we use a focused fragment of the Scenic traffic situation modeling language [8].

III. Representing models with EMF and VIATRA

First, we present the domain of measurement using standard modeling technologies.

First, review the metamodel. A metamodel describes the main concepts and relations, of a model, and defines its main structure. In this paper, we used a simple metamodel shown in Figure 1 using EMF. In this metamodel, a Traffic situation consists of Lanes and Cars. A Lane can be connected to another Lane via the following reference, and its relation with the other lanes are represented with the left and right references. Cars are placed onto a single lane. Instance models in our measurements are in accordance with this metamodel

In our measurements we are using VIATRA as a comparison. We implemented seven graph patterns in VQL to query

- incoming empty lane segments without preceding lanes to spawn new cars into the scenario;
- outgoing lane segments without following lanes to despawn cars from the scenario;
- cars on the same or adjacent lanes for listing potentially dangerous situations;
- and potential trajectories for lane following and changing maneuvers [8].

The implementation of the last pattern is illustrated below.

```
pattern moveCar(from: Lane, to: Lane, car: Car) {
    Lane.following(from,to);
    Car.on(car,from); }
or{ Lane.left(from,to);
    Car.on(car,from); }
or{ Lane.right(from,to);
    Car.on(car,from); }
```

IV. 4-valued partial models

Next, we illustrate the same problem with 4-valued partial models. Partial modeling is a technique to explicitly represent uncertainty in models by abstracting a collection of possible models into a single partially specified model. In this paper, we use 4-valued logic to represent uncertainty, where traditional

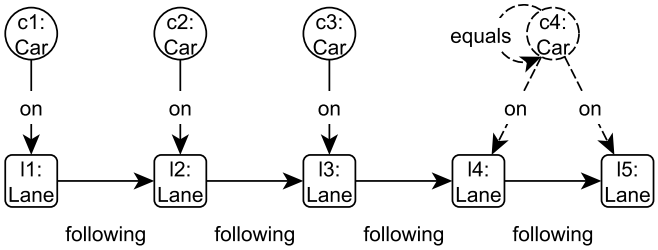


Fig. 2: Partial model with unknown relation values

Car	
c1	**true**
c2	**true**
c3	**true**
c4	**true**

Following		
l1	l2	**true**
l2	l3	**true**
l3	l4	**true**
l4	l5	**true**

exists	
l1	**true**
···	**true**
c3	**true**
c4	**unknown**

Lane	
l1	**true**
l2	**true**
l3	**true**
l4	**true**
l5	**true**

On		
c1	l1	**true**
c2	l2	**true**
c3	l3	**true**
c4	l4	**unknown**
c4	l5	**unknown**

equals		
l1	l1	**true**
···		**true**
c3	c3	**true**
c4	c4	**unknown**

TABLE I: Relational representation of an example model

logic values **true** and **false** are extended with a logic value **unknown** to represent uncertain or incomplete data where both true and false are possible in the represented concrete models, and with **error** to represent inconsistencies.

Figure 2 shows a partial model with five lanes and four cars. Lanes 1-5 are following each other, and cars 1-3 are on lanes 1-3. We know that these objects and relations exist, so they are marked with solid arrows. The last car, c4 can be on lanes l1 or l2. In this example, it is unknown whether it is on either of them, these relations are marked with dashed arrows. We extend this notation of uncertainty to existence and equivalence as well, which enables abstract nodes that can represent multiple or no nodes. In our example, c4 is denoted with a dashed loop edge with the label "equals", which means that c4 can represent multiple nodes. By default, other objects are different from each other, and equal with only themselves. Moreover, c4 is denoted with dashed line, which means that its existence is uncertain: it can be included to, or excluded from the model.

The same model is shown in Table I, in similar tables as the ones used in our predicate evaluation algorithm. The Car and Lane tables show the types of the objects, while the Following and On tables contain the relations between the objects. A **true** value means the relation exists in the model, while an **unknown** value means the relation may exist in the model. The most numerous **false** values are omitted in both Figure 2 and Table I. Error values are not present in the model, since that would mean there is an inconsistency.

4-valued partial models enable the representation of classes and references with unknown existence using abstract nodes and edges like c4 in Figure 2. As a syntactic sugar, [7] introduces classes and references (illustrated below) which are translated to abstract nodes and edges internally.

```
class Lane {                    class Car {
```

```
  Lane[0..*] following                Lane[0..1] on
  Lane[0..1] left opposite right    }
  Lane[0..1] right opposite left
}
```

The predicates implementing the graph patterns in the new specification language are shown below. We chose to use direct predicates (**direct pred**), so we could specify to match for **true** and **unknown** values for potential values of car trajectories, giving us a 2-valued result. Using a predicate (**pred**) without specifying the **true** and **unknown** values would give a 4-valued result [9]. These predicates are semantically equivalent to the ones implemented in VIATRA.

```
direct pred moveCar(from,to,car) <->
   following(from,to)=true|unknown,on(car,from)=true|unknown
 ; left(from,to)=true|unknown,on(car,from)=true|unknown
 ; right(from,to)=true|unknown,on(car,from)=true|unknown.
```

V. Evaluation

A. Research questions

We evaluated the performance of the query engine by formulating various research questions and answering them by measuring execution times. These are the research questions we aim to answer:

RQ2 How does the model building scale if we increase the model size?

RQ1 How does the pattern matching scale if we increase the model size?

B. Measurement setup

The measurement workflow is shown in Figure 3. The measurements have three parameters:

- x: The number of lanes following each other.
- y: The number of parallel lanes.
- n: The number of times the changes are applied and pattern matching is executed.

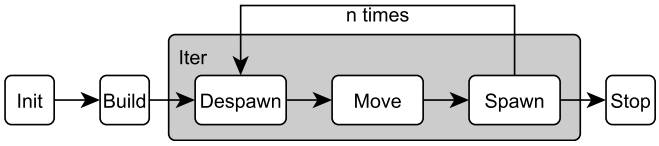


Fig. 3: Measurement setup

First, we initialize the predicates, an empty model and the query engine in the Init phase. Next, in the Build step we build up the model, which consists of $x * y$ lanes in a grid, and y cars placed randomly on these lanes. Figure 4 shows an example of a four by four grid of lanes with four cars. The forward direction is to the right, and the arrows show which lane each car is able to move to.

In the next step an iteration starts, where we first despawn all cars that are on a lane that has no following lane. In Figure 4 car C4 would be removed, since it can no longer go forward. Next we move all remaining cars from their current lane to its following lane or either of the lanes next to them. After this step the example might look like Figure 5. The third and final step in the iteration is spawning new cars to make sure there are y cars in the model. These new cars are placed randomly on the lanes. These three steps are repeated n times. After n iterations the measurement stops. The runtime of each step is measured separately.

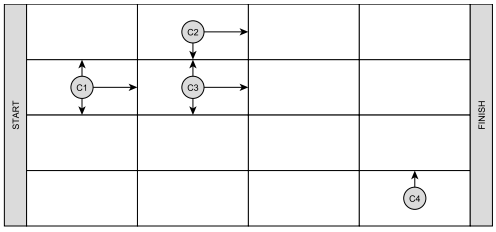


Fig. 4: Example model of a 4×4 grid of lanes with 4 cars

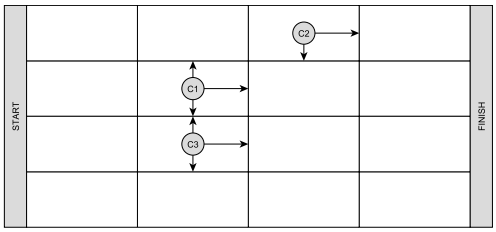


Fig. 5: An example model after despawning and moving

C. Compared approaches

We ran the same measurement using four different approaches. First, we measured VIATRA in continuous evaluation mode (where each model change is processed immediately, denoted by VQL-continuous), then in coalescing mode (where model updates are processed after the full iteration, denoted by VQL-batch. VIATRA does not support 4-valued evaluation naively, those are matched on only **true** and **false** values, using EMF as data structure. Then we measured our approach once where we query both **true** and **unknown** values (where all four logic value is used), and once where we query exactly **true** values (denoted with Refinery and Refinery-Abstract). We run the simulation for 5000 iterations, while saving the runtime after every 1000 iterations. Before the measurement of both tools, we ran a similar, but smaller setup to account for the JVM warm up, and programmatically called the garbage collector after each run. Each measurement was repeated 25 times, and we used the median value of the results, to filter out the noise.

We executed the measurements for multiple model sizes. The sizes are illustrated in Table II For the measurements we used the following hardware, software versions, and settings: Java version: 17, maximum Java heap size: 8GB, VIATRA version: 2.6.0, OS: Windows 10, CPU: Intel Core i7-9750H.

D. Measurement results

Figure 6a shows the runtime building the model. The horizontal axis is the nodes and edges in the model. The vertical axis shows the time it took to complete building the model, in milliseconds.

size	lanes	cars	nodes+edges
50	50x50	50	10100
100	100x100	100	40200
150	150x150	150	90300
200	200x200	200	160400
250	250x250	250	250500
500	500x500	500	1001000
750	750x750	750	2251500
1000	1000x1000	1000	4002000
1250	1250x1250	1250	6252500

TABLE II: Model sizes

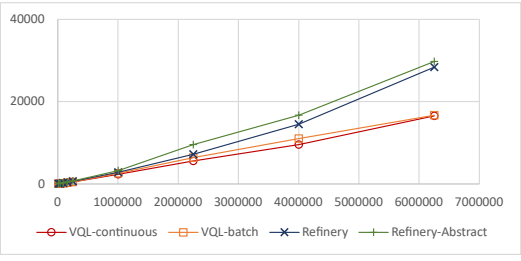
Figure 6b shows how the runtime changes if we increase the number of nodes and edges in the model used in the setup. The horizontal axis is the size of the model on which we ran the pattern matching, as detailed above. The vertical axis shows the time it took to complete 5000 iteration of modification on the model, in milliseconds.

E. Discussion of the results

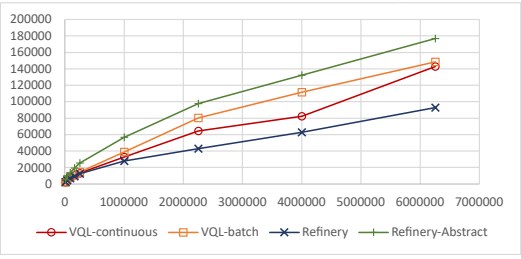
On Figure 6a we can see that building the model is slower with our solution, than the two VIATRA configuration, because the current version of Refinery and Refinery-Abstract uses more relation (both left and right, exists and equals), and needs twice as many base indexing to support multiple logic values. However, these initial performance measurement showed potential performance improvements.

RQ1 With respect to the model size the original VIATRA scales better than Refinery and Refinery-Abstract.

As we can see in Figure 6b, all four measurements show a similar shape on the diagram. The fastest solution was our approach without abstraction (Refinery), and provided better performance then both VQL-continuous and VQL-batch are slower. This can happen as it uses more advanced data structures than EMF and skips model management steps irrelevant to running the core query evaluation engine (e.g., notification sending, order of elements in a list, resource management). The slowest solution was our approach with abstraction, where the result is calculated from the combination of predicates. This took almost twice as long to run, than without abstraction, since this had to check roughly twice as many rows.



(a) Runtime of model building



(b) Change in total runtime by model size

Fig. 6: Runtime measurements

RQ2 With respect to the model size, Refinery scales better than the original VIATRA. Compared to that, Refinery-Abstract needs almost twice as much time.

VI. Conclusion and Future Work

In this paper, we provided an initial performance benchmark using VIATRA with a novel data structure representing 4-valued partial models. Despite the richer expression power of our data structure, our solution produced favorable performance, better than the standard modeling technology (EMF).

In the future, we are planning to use this data structure in a design space exploration scenario used for model generation, replacing the backed engine of VIATRA Solver. Additionally, we are planning to evaluate the performance of the data structure using existing benchmarks (like [10]).

Acknowledgements: The first author was partially supported by the European Commission and the Hungarian Authorities (NKFIH) through the Arrowhead Tools project (EU grant agreement No. 826452, NKFIH grant 2019-2.1.3-NEMZ ECSEL-2019-00003) and by ÚNKP-21-4 New National Excellence Program of the Ministry for Innovation and Technology from the source of the National Research, Development and Innovation Fund. The second author was partially supported by the NRDI Fund of Hungary, financed under the [2019-2.1.1-EUREKA-2019-00001] funding scheme.

References

[1] O. Semeráth, A. S. Nagy, and D. Varró, "A graph solver for the automated generation of consistent domain-specific models," in *Proceedings of the 40th International Conference on Software Engineering*, pp. 969–980, 2018.

[2] G. Bergmann, Z. Ujhelyi, I. Ráth, and D. Varró, "A graph query language for EMF models," in *Theory and Practice of Model Transformations*, pp. 167–182, Springer Berlin Heidelberg, 2011.

[3] H. Abdeen, D. Varró, H. Sahraoui, A. S. Nagy, C. Debreceni, Á. Hegedüs, and Á. Horváth, "Multi-objective optimization in rule-based design space exploration," in *ACM/IEEE international conference on Automated software engineering*, pp. 289–300, 2014.

[4] G. Soltana, M. Sabetzadeh, and L. C. Briand, "Practical constraint solving for generating system test data," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 29, no. 2, pp. 1–48, 2020.

[5] M. Richters and M. Gogolla, "OCL: Syntax, semantics, and tools," in *Object Modeling with the OCL*, pp. 42–68, Springer, 2002.

[6] M. J. Steindorfer and J. J. Vinju, "Optimizing hash-array mapped tries for fast and lean immutable jvm collections," *SIGPLAN Not.*, vol. 50, p. 783–800, oct 2015.

[7] K. Marussy, O. Semeráth, A. A. Babikian, and D. Varró, "A specification language for consistent model generation based on partial models," *J. Object Technol.*, vol. 19, no. 3, pp. 3:1–22, 2020.

[8] D. J. Fremont, T. Dreossi, S. Ghosh, X. Yue, A. L. Sangiovanni-Vincentelli, and S. A. Seshia, "Scenic: a language for scenario specification and scene generation," in *ACM SIGPLAN Conference on Programming Language Design and Implementation*, pp. 63–78, 2019.

[9] O. Semeráth and D. Varró, "Graph constraint evaluation over partial models by constraint rewriting," in *International Conference on Theory and Practice of Model Transformations*, pp. 138–154, Springer, 2017.

[10] G. Szárnyas, B. Izsó, I. Ráth, and D. Varró, "The train benchmark: cross-technology performance evaluation of continuous model queries," *Software & Systems Modeling*, vol. 17, no. 4, pp. 1365–1393, 2018.

Application of Coherence Function to the Analysis of Compressive Sensing

András Palkó, László Sujbert
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: {palko, sujbert}@mit.bme.hu

***Abstract*—Compressive sensing has been developed for the sampling of sparse or compressible signals. Strong theorems state that when a signal is sufficiently sparse, its samples can be accurately recovered from random sub-Nyquist measurements. As a consequence, compressive sensing is emerging as a part of various applications, such as image processing, biomedical problems or audio signal processing. Designing a compressive sensing application comprises the selection of many parameters, e.g. data acquisition scheme, compression ratio, reconstruction algorithm, etc. To make these decisions experimentally, a simple criterion to compare several options can prove to be helpful. This paper proposes to use the coherence function as a criterion to evaluate the quality of a signal transmission via compressive sensing. After a brief review of compressive sensing, the usage of the coherence function is presented. Simulation examples illustrate how it can help making the design decisions.**

***Index Terms*—coherence function, compressive sensing, FFT, stochastic signals**

I. Introduction

Traditionally, sampling is governed by Shannon's theorem. This well-known result is universal, it can be used for sampling any signal. In practice, many signals can be described with only a few significant coefficients in an appropriate basis, frame or dictionary (for brevity, in the following only the word basis will be used). This phenomenon is called sparsity.

A signal is sparse if there is a basis in which it has few nonzero coefficients. Similarly, a signal is compressible in a basis if its sorted coefficients decay rapidly (enveloped by an exponential decay). Whether a signal is sparse (compressible) or not, depends on the basis. To illustrate this, one can consider the (inverse) discrete Fourier transform of a single spike. A basis in which a signal has a sparse representation, is called the sparsifying basis (for that signal).

Compressive sensing was introduced in 2004 by Donoho, Candès, Romberg and Tao [1], [2], [3] for the sampling of sparse or compressible signals. Traditionally, using Shannon's theorem, one would take a number of samples, and then use a compression algorithm to represent the signal with a fewer number of samples. Compared to the sparsity of the signal, one oversamples it, then performs the compression and only keeps the significant coefficients. Thus, a great part of the acquired data is discarded. In contrast, using compressive sensing, one directly obtains a compressed representation via random sampling. Sampling and compression are performed simultaneously, at a sub-Nyquist rate.

This research was funded by the National Research, Development, and Innovation Fund of Hungary under Grant TKP2021-EGA-02.

One would expect that if the sampling is sub-Nyquist, the signal cannot be reconstructed exactly. For general signals, this is true. However, for sparse signals compressive sensing offers accurate reconstruction from sub-Nyquist measurements using nonlinear reconstruction algorithms [4], [5], [6].

Applications of compressive sensing are emerging in various fields of science and technology. A famous image processing example is the single-pixel camera [7]. Some other fields are biomedical problems [8], or face recognition [9]. For a broad overview of compressive sensing acquisition and reconstruction strategies, as well as applications we refer the readers to the survey paper [10].

When one designs an application of compressive sensing, there are multiple decisions to make. A major task is to determine the sparsifying basis. Moreover, one needs to decide the data acquisition and the reconstruction schemes. They can have several parameters to tune, the most trivial is the rate of compression. These decisions may require extensive knowledge about the compressive sensing structures and algorithms.

Another way of making these design decisions is via experimentation. In many applications, the signals to be processed can be modeled well by stochastic signals, e.g. noise or vibration signals; nonstationary signals; audio, acoustic and speech signals or signals containing short periodic parts. Furthermore, in many applications, the signals' frequency domain behavior is technically relevant. In these cases, it is important to accurately transmit those frequency bands which contain the signal. When such stochastic signals are transmitted through a system, the transmission quality can be assessed in the frequency domain by calculating the coherence function between the input and output signals. We propose to use the coherence function in order to help making the design decisions by experimentation.

The paper is arranged as follows: Section II gives an overview of compressive sensing. The usage of the coherence function is discussed in Section III. Section IV presents some simulation examples. The paper concludes in Section V.

II. Compressive Sensing

Compressive sensing can be split into two tasks:

- Data acquisition: getting the compressed measurements from the input signal.
- Reconstruction: getting the estimate of the input signal from the compressed measurements.

In the following, these tasks are reviewed briefly.

A. Data Acquisition

Data acquisition can be modeled as follows:

$$y = \varphi x \tag{1}$$

where $x \in \mathbb{C}^n$ is the input vector, $\varphi \in \mathbb{C}^{m \times n}$ is the measurement matrix and $y \in \mathbb{C}^m$ is the vector of compressive measurements.

Usually φ is chosen as a random matrix, e.g. with elements drawn from a Gaussian distribution. Furthermore, $m = \tau n$, where $0 < \tau \leq 1$ is the compression ratio [11]. $\tau < 1$ means $m < n$. In certain applications, even $m \ll n$ can be achieved. The actual value of τ depends on the sparsity of the signal. To obtain a low value, the measurement matrix needs to be incoherent with the sparsifying basis [12].

B. Reconstruction

In the reconstruction problem, we are given the measurements y, the measurement matrix φ, and we try to solve the measurement equation (1) for x. This is an underdetermined system with infinitely many solutions. The usual least squares approach yields poor results, since it tries to give a solution with minimal energy, disregarding the sparsity of the signal.

If $\psi \in \mathbb{R}^{n \times k}$ is the sparsifying basis of the signal, that is

$$x = \psi s \tag{2}$$

where $s \in \mathbb{R}^k$ (or $\mathbb{C}^k$), $k \geq n$ is the sparse (or compressible) coefficient vector, then the measurement equation (1) can be rewritten as

$$y = \varphi \psi s = \Theta s. \tag{3}$$

Since we know that s is a sparse vector, the unique solution of the reconstruction problem can be determined by choosing the sparsest possible s. Mathematically, this is described by the following l_0 optimization:

$$\hat{s} = \arg\min_s \|s\|_0 \quad \text{subject to} \quad y = \Theta s \tag{4}$$

where $\hat{s}$ is the estimated coefficient vector and $\|\cdot\|_0$ denotes the l_0 pseudonorm which is the number of nonzero elements.

Now the input signal can be estimated:

$$\hat{x} = \psi \hat{s}. \tag{5}$$

Directly solving (4) is computationally extensive, since it involves trying all the possible combinations, which is an NP-hard problem. Several alternate methods have been proposed in the literature, e.g. to use convex optimization (l_1 norm):

$$\hat{s} = \arg\min_s \|s\|_1 \quad \text{subject to} \quad y = \Theta s \tag{6}$$

This modified reconstruction problem can be solved using linear programming techniques. This solution is called Basis Pursuit [4].

III. Coherence Function

The $\gamma^2(f)$ coherence function [13] between signals x and z is defined as

$$\gamma^2_{xz}(f) = |S_{xz}(f)|^2 / (S_{xx}(f) S_{zz}(f)) \tag{7}$$

where $S_{xx}(f)$ and $S_{zz}(f)$ are the auto power spectral densities, while $S_{xz}(f)$ is the cross power spectral density. In practice, they can be efficiently estimated using FFT. Note that this coherence function is a different concept than the (in)coherence between the φ measurement and the ψ sparsifying matrices [14] used in the field of compressive sensing.

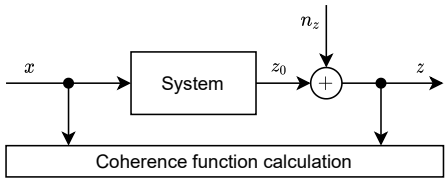


Fig. 1. Setup used for the coherence function example

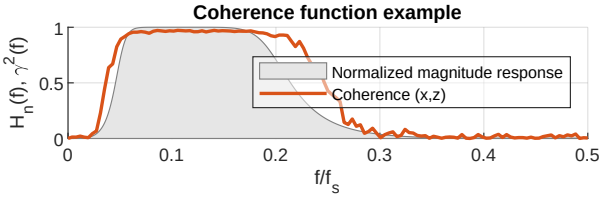


Fig. 2. Coherence function example. Grey area: normalized magnitude response, $|H|_{\max} = 1$. Red line: coherence between signals x and z.

At each frequency, the coherence function indicates the correlation between signals x and z. That is, $0 \leq \gamma^2_{xz}(f) \leq 1$ and a high value indicates a linear relationship between signals x and z. The coherence is decreased in the presence of uncorrelated noise or nonlinearities in the system.

The following demonstrative example illustrates the application of the coherence function for assessing the quality of a signal transmission.

Fig. 1 depicts the setup: first, 10000 samples of Gaussian white noise are generated (signal x). Then, it is processed by a 4th order Butterworth bandpass filter (passband: $[0.05 \ldots 0.2] \cdot f_s$, where f_s is the sampling frequency) to obtain z_0. Then the n_z Gaussian white noise is added to the output with SNR $= 10$ dB to get the z output signal. Finally, the coherence function is calculated for the input against the noisy output.

The coherence function is shown in Fig. 2 alongside the $H_n(f) = |H(f)| / |H|_{\max}$ normalized magnitude response of the system ($|H|_{\max}$ is the maximal magnitude response). The coherence between the noisy output and the input is drawn with red color, while the gray area is the magnitude response of the filter. To make the coherence function easily comparable to the magnitude response, the magnitude response is drawn as an area, on a linear scale.

The coherence between the noisy output and the input signal is as one would expect: it is high (low) when the magnitude response is high (low). Since the spectrum at the input x is white, the spectrum of z_0 is shaped like the magnitude response. Because of the noise n_z, the output z is dominated by the noise in the stopband, where the magnitude response is low. As this noise is uncorrelated with the input, it is not contained in the $S_{xz}(f)$ cross spectrum, but contained in the $S_{zz}(f)$ output spectrum. Thus, the numerator of (7) is unchanged, while its denominator grows: the coherence is decreased. Similar arguments can be made for the passband.

A potential application of compressive sensing is compressing the transmitted data. In many cases, the signals to be transmitted can be modeled well by stochastic signals. Some examples are noise measurement [15], vibration analysis [16],

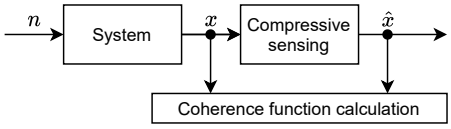


Fig. 3. Setup used for the simulations

or signals containing short periodic parts such as speech [17], acoustic or in general audio signals.

In these cases, the transmission utilizing compressive sensing can be modeled as the processing of a stochastic signal with a system. For this model, a possible way of assessing the quality of the transmission is to calculate the coherence function between its input and output signals. We propose to use the coherence function in order to help making the design decisions by experimentation.

A. Alternative Metrics

There are several metrics to evaluate a compressive sensing scheme [11]. E.g. two popular ones are the Normalized Root Mean Square Error (NRMSE) and the Signal-to-Error Ratio (SER):

$$\mathrm{NRMSE} = \|x - \hat{x}\|_2 \,/\, \|x\|_2 \tag{8}$$

$$\mathrm{SER} = -20 \lg \mathrm{NRMSE} \tag{9}$$

Note that while the NRMSE is a normalized metric, it can obtain values higher than one (since the normalization refers to the division by $\|x\|_2$).

The usual metrics give a scalar, integral measure of the quality from time domain analysis. In contrast, the coherence function provides a vector of quality metrics in the frequency domain.

IV. Examples

To illustrate how the coherence function can help designing a signal transmission using compressive sensing, some simulation examples are presented.

Thus the aim of the examples below is not to illustrate the power of an optimized data transmission using compressive sensing, but to present how the coherence function shows the difference between various compressive sensing schemes. As a consequence, not necessarily the best sparsifying bases or the most powerful reconstruction algorithms are used.

A. The Simulation Environment

The setup used for the simulations is shown in Fig. 3. To generate the x input signal, the n Gaussian white noise is processed with a system. Then this x signal is passed through the compressive sensing data acquisition–reconstruction scheme to obtain the $\hat{x}$ estimate of the input signal.

The elements of the φ measurement matrix are drawn from a Gaussian distribution. The τ compression ratio was varied from 5% to 50% in steps of 5%. The l_1-magic implementation [18] of solving (6) is used for the reconstruction. For simplicity, only changes in τ and in the sparsifying basis are shown in the examples. However, the proposed analysis is applicable e.g. to changes in the reconstruction algorithm as well.

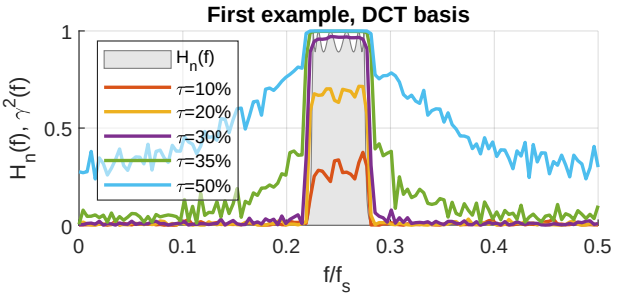


Fig. 4. First example, DCT basis. Grey area: normalized magnitude response, $|H|_{\max} = 1$. Colored lines: coherence function at different compression ratios.

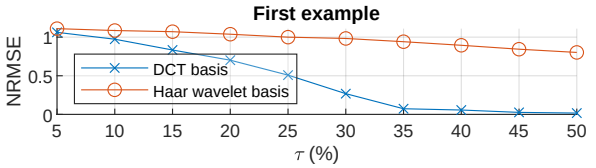


Fig. 5. NRMSE with different compression ratios in the first example

B. Example 1: Elliptic Bandpass Filter

In the first example, the system is an elliptic bandpass filter (6th order, 1 dB passband ripple, 60 dB stopband attenuation, $[0.45 \ldots 0.55] \cdot f_s/2$ passband). Its magnitude response is shown in Fig. 4 as the grey area. Consequently, the signal x should contain significant coefficients only in the passband, that is, x should be approximately sparse.

The discrete cosine transform (DCT) [19] is selected for the sparsifying basis as an initial choice. The results are shown in Fig. 4 with the colored lines. As the compression ratio increases, the coherence increases first in the passband, then also in the stopband. The coherence function takes almost 1 values in the passband at $\tau \geq 35\%$. From this, one could infer that $\tau = 35\%$ is enough for a good transmission.

This claim can be verified by looking at the NRMSE. Fig. 5 shows this error for the investigated τ values. The error decreases rapidly from 30% to 35%, while from 35% to 50% it is still decreasing, but at a slower rate. There is a clear break in the NRMSE where the coherence function fully "envelopes" the input spectrum.

From the perspective of the designer, plotting the input spectrum and the coherence function is more informative than calculating a single number.

Other potential sparsifying bases can be tried and evaluated, here the results of the Haar wavelet basis [20] are presented (Fig. 6). Comparing to Fig. 4, it is clear that the coherence at a given τ is worse for the Haar wavelet basis than for the DCT basis. This also can be seen on the NRMSE plot. Consequently, the DCT basis is a better choice than the Haar wavelet basis in this example.

Note that for both bases, the coherence function has higher values in the passband and lower values in the stopband. This is as expected: as τ grows, less and less significant parts of the signal are getting transmitted also. When the coherence is high in all the significant bands, the useful information in the signal are transmitted. This is harder to see in the time domain error plot.

A potential remark would be the idea to transform this

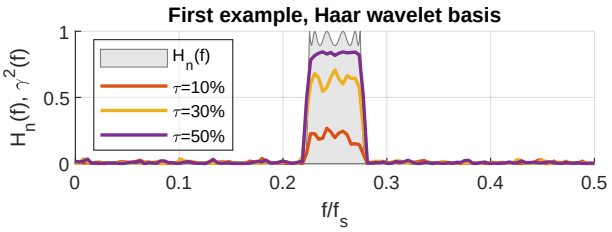


Fig. 6. First example, Haar wavelet basis. Gray area: normalized magnitude response, $|H|_{\max} = 1$. Colored lines: coherence function at different compression ratios.

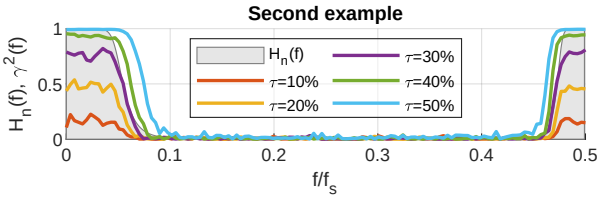


Fig. 7. Second example, DCT basis. Grey area: normalized magnitude response, $|H|_{\max} = 1$. Colored lines: coherence function at different compression ratios.

signal to baseband and there use traditional sampling. In this simple case, this is a viable solution. However consider a case when the signal has several bands scattered in the spectrum, with the same "total bandwidth" as here. In such a case, compressive sensing requires similar compression ratio as here, while generally the transformation to baseband is not applicable.

C. Example 2: Butterworth Bandstop Filter

The system in the second example is a Butterworth bandstop filter (6th order, $[0.1 \ldots 0.95] \cdot f_s/2$ stopband). Fig. 7. illustrates its normalized magnitude response with the gray area. In the first example, there was a single, narrow passband. Here, the passband is still narrow, but is split into two bands. Similarly to the first example, x should be approximately sparse.

After calculating the coherence function with the DCT basis, we got the results shown in Fig. 7. Compared to the previous example, at first glance now we can see that $\tau = 50\%$ is required to reach the coherence value of 1 in the passbands. This is larger than there, however, the total width of the passbands is also larger than in the first example.

The time domain analysis shows that $\tau = 50\%$ transmission offers similar quality to the $\tau = 35\%$ case in the first example (Fig. 8). Again, the coherence is higher in the passbands and lower in the stopband.

V. Conclusion

In this paper the usage of coherence function was proposed to assess the transmission quality of stochastic signals via compressive sensing. After taking an overview of the compressive sensing process, the coherence function was reviewed. In many signal processing applications, the signals' frequency domain behavior is technically relevant. Thus, it is important to accurately transmit those frequency bands which contain the majority of the signal's power. When such stochastic signals are transmitted through a system, the coherence function can be used as a tool to compare the quality of different data transmission options. Simulation examples illustrated the usage of coherence function to qualify a signal transmission via compressive sensing. The results showed that in certain simple cases, similar compression can be reached with compressive sensing as with traditional methods. A potential future task is finding such examples which can better illustrate the usage of the coherence function.

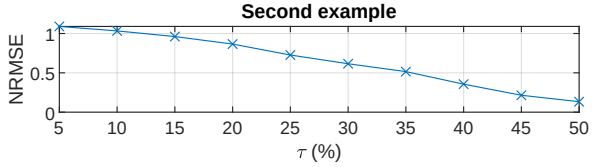


Fig. 8. NRMSE with different compression ratios in the second example

References

[1] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[2] D. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.

[4] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, 1998. [Online]. Available: https://doi.org/10.1137/S1064827596304010

[5] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. [Online]. Available: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/j.2517-6161.1996.tb02080.x

[6] D. Needell and J. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1063520308000638

[7] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 83–91, 2008.

[8] K. Poh and M. Pina, "Compressive sampling of eeg signals with finite rate of innovation," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, pp. 1–13, 02 2010.

[9] P. Nagesh and B. Li, "A compressive sensing approach for expression-invariant face recognition," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1518–1525.

[10] M. Rani, S. B. Dhok, and R. B. Deshmukh, "A systematic review of compressive sensing: Concepts, implementations and applications," *IEEE Access*, vol. 6, pp. 4875–4894, 2018.

[11] F. Salahdine, E. Ghribi, and N. Kaabouch, "Metrics for evaluating the efficiency of compressing sensing techniques," in *2020 International Conference on Information Networking (ICOIN)*, 2020, pp. 562–567.

[12] R. Baraniuk, "Compressive sensing [lecture notes]," *IEEE Signal Process. Mag.*, vol. 24, pp. 118 – 121, 08 2007.

[13] J. S. Bendat and A. G. Piersol, *Random Data: Analysis and Measurement Procedures, 4th Edition*. Wiley Series in Probability and Statistics, Feb. 2010.

[14] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, 2008.

[15] J. Su, H. Zhu, R. Mao, C. Su, and L. Guo, "A method to identify the noise radiation regions of acoustic source," in *2016 IEEE/OES China Ocean Acoustics (COA)*, 2016, pp. 1–5.

[16] R. Potter, "A new order tracking method for rotating machinery," *Sound and Vibration*, vol. 24, no. 9, pp. 30–34, 1990.

[17] L. Bu and T.-D. Church, "Perceptual speech processing and phonetic feature mapping for robust vowel recognition," *IEEE Trans. Speech Audio Process.*, vol. 8, no. 2, pp. 105–114, 2000.

[18] E. J. Candès and J. Romberg, "l1-magic: Recovery of sparse signals via convex programming," Oct 2005, accessed: 2021-12-05. [Online]. Available: https://candes.su.domains/software/l1magic/

[19] K. R. R. Vladimir Britanak, Patrick Yip, *Discrete Cosine and Sine Transforms*. Elsevier, Sept. 2006.

[20] P. J. V. Fleet, *Discrete Wavelet Transformations: An Elementary Approach with Applications*. John Wiley & Sons, Jan. 2008.

Approximate time-optimal model predictive control of a SCARA robot: a case study

Bence Cseppentő[1,2], Jan Swevers[1], Zsolt Kollár[2]
[1]KU Leuven, Department of Mechanical Engineering
DMMS-M Core Laboratory, Flanders Make
Leuven, Belgium; Email: jan.swevers@kuleuven.be
[2]Budapest University of Technology and Economics, Department of Measurement and Information Systems
Budapest, Hungary; Email: {cseppento, kollarzs}@mit.bme.hu

***Abstract*—This paper investigates, based on a case study of a SCARA robot, how time-optimal point-to-point motion can be approximately realized using a model predictive control formulation that has low computational complexity. Time-optimality is realized by an indirect formulation and different objective functions are compared. Using the ℓ_1-norm instead of a quadratic penalty mimics time-optimality, however, to reduce computational complexity and evade some of its disadvantages, the Huber-norm is introduced as a velocity penalty while the position is penalized quadratically. The contribution of the sampling rate and the length of the prediction horizon is also examined, as the sampling rate poses a limit on the available computation time, while the prediction horizon influences computational complexity. Simulations were carried out in a MATLAB environment using the CasADi toolbox.**

***Index Terms*—optimal control, time-optimal motion, PTP motions, SCARA robot, CasADi, Huber-norm**

I. Introduction

Model predictive control (MPC) methods have been a topic of research for decades. With the advancement of digital technology it is becoming more and more economic to implement sophisticated nonlinear programming techniques for online controllers in many fields [1].

Although there is immense literature about MPC techniques [1] and optimization [2] [3], the choice of sampling frequency and the size of the finite prediction horizon pose a challenge as there are a number of issues to consider. Firstly, the sampling time has to be larger than the computation time of the next control action. Secondly, the computation time is affected by the number of samples in the prediction window as well as by the number of integration intervals within one sample. Thirdly, for better performance a large enough window is needed, while the integration interval has to be small enough to accommodate the system dynamics. Thus, a compromise must be reached.

The aim of this paper is to investigate the effect of sampling rate and prediction horizon size on controller performance, while examining different objectives to mimic time-optimal motion with low computational complexity. The plant considered in this study is a rigid-body forward dynamics model of an EPSON G3-251 SCARA robot [4] [5] [6]. This robot has two rotational joints and one prismatic joint, the latter of which is not used in the presented work. The simulations were carried out on a forward dynamics model, i.e. the states of the plant are the joint positions (rotational angles), $\boldsymbol{q}$ and the joint velocities, $\dot{\boldsymbol{q}}$, while the inputs are the actuator torques, $\boldsymbol{\tau}$.

The rest of the paper is organized as follows: Section II elaborates upon the implementation and parameters of the simulations. Section III illustrates the examined performance indicators for various sampling rates, prediction horizon windows and objective functions. Finally, Section IV summarizes the conclusions and presents possible future work.

II. Simulation setup

The algorithm used in this paper iteratively solves a constrained nonlinear optimization problem over a finite time horizon. The calculated input corresponding to the first sampling interval of the prediction horizon is applied to the model of the system, and the resulting state variables are the new initial values for the repetition of the process.

The algorithm has been implemented using MATLAB and the CasADi tool [7] supplemented by the CasADi-reliant Rockit (Rapid Optimal Control kit) [8]. To solve the nonlinear program, the COIN-OR Ipopt interior point optimization solver is used [9]. The forward dynamics of the plant is available in the form of an ordinary differential equation (ODE) as a CasADi function, which is discretized by Rockit using a 4^{th} order Runge-Kutta integrator with one internal stage. After solving the optimal control problem (OCP), the first element of the resulting control sequence is applied to a zero-order hold discrete-time equivalent of the forward robot model to evaluate the resulting state variables, which are used as initial values for the next OCP solve. For now, all simulations were carried out without considering model inaccuracies or noise.

A. Motion scenarios

The robot has two rotational joints; a shoulder and an elbow joint. After evaluating a number of different scenarios and analyzing the data, it became apparent that two cases may be distinguished: the range of motion may be large or small relative to the sampling time and prediction horizon. The presented data will concern two extreme cases in order to illustrate the importance of the examined parameters:

- both the shoulder and elbow joints utilize 95% of their range, presenting a scenario of a large-range motion

DOI 10.3311/MINISY2022-003

- only the elbow joint rotates by 6°, presenting a small motion.

B. Parameters of the simulation

Besides the initial and final positions, two important, adjustable parameters are observed: the sampling frequency (f_s) which defines the sampling time ($T_s = \frac{1}{f_s}$) and the number of control intervals in the prediction horizon (N_control).

Note that the prediction horizon is not defined as a time window. If it were, because the window size is $N_\mathrm{control} \cdot T_s$, increasing f_s would mean proportionally increasing N_control, leading to an increase in computation time as well. Hence, for each f_s the same set of N_control values were used.

C. Constraints of the OCP

The OCP to be solved for each control action is constrained by the physical bounds of the joint angles and the maximum attainable values of the angular velocities and actuator torques.

D. Objective function of the OCP

The objective function to be minimized is a combination of running costs on the position with respect to the desired final position and the angular velocities. Although traditionally MPC designers choose quadratic cost functions, there is a growing tendency to exploit linear penalties in order to achieve near time-optimal motion [10] [11]. For a time-optimal point-to-point motion, aggressive control action is needed in order to accelerate as fast as possible to the maximum velocity, keep that speed constant as long as possible, then decelerate as fast as possible to be able to stop at the final position. In this study a combination of different cost functions is used: as a quadratic penalty weighs large cost function values higher, this is imposed on the joint position, while the effect of different velocity penalties is compared.

In the literature, besides near time-optimality, using the ℓ_1-norm is further motivated by reducing complexity due to the optimization becoming a linear program instead of a quadratic one; however, in this case study the system dynamics is nonlinear and the position penalty remains quadratic. It also introduces some drawbacks. As the ℓ_1-norm is a non-differentiable function, in implementation a slack variable is introduced as the objective, and an additional inequality constraint ensures the connection between the angular velocities and the slack variable. Because of this, the solution of the OCP might take more time. Furthermore, the ℓ_1-norm may introduce deadbeat control behavior, potentionally causing stability issues, especially for short prediction horizons. For these reasons the Huber-norm is introduced, which is quadratic for small, while linear for large velocity values. Besides switching to a quadratic penalty towards the end of the motion to ensure stability, another advantage is that it can be approximated by a smooth function called the pseudo-Huber norm:

$$\ell_{\delta,k} = \delta^2 \left(\sqrt{1 + \left(\frac{\epsilon_k}{\delta}\right)^2} - 1 \right); \epsilon_k = \left\| \boldsymbol{q}_{k+1} \right\|_2 \qquad (1)$$

where $\boldsymbol{q}_{k+1}$ is the vector of angular velocities at time index k, while δ is a new parameter defining the velocity norm at which the transition between linear and quadratic penalty is made.

The Huber-norm may also be implemented by introducing two slack variables. Although three additional constraints increase complexity, this has been tested as well.

III. Evaluation of Results

In order to evaluate the results, some performance metrics must be defined. In this study these were the settling time ($\pm 5\%$) and the statistics of the number of iterations required per control action (mean, worst case and standard deviation).

The data presented comprise four f_s values: $\{250\mathrm{Hz}; 500\mathrm{Hz}; 750\mathrm{Hz}; 1000\mathrm{Hz}\}$, and nine N_control values: $\{3; 4; 5; 6; 7; 8; 9; 10\}$. In the cases using the Huber-norm, for the large motion $\delta = \{5\}$; for the small motion $\delta = \{1; 0.5\}$.

A. Settling time

Fig. 1 illustrates how the different velocity penalties affect the settling time for the large motion range scenario.

The immediately noticeable main difference is that when using a quadratic velocity penalty, the prediction horizon required for minimal settling time is larger than in the other cases as the sampling rate increases. The settling time is practically independent of N_control for the other norms. Hence, from the perspective of time-optimality the linear penalty is superior to the quadratic one, while a suitable choice of the δ parameter of the Huber-norm achieves the same settling time even if the prediction horizon contains sufficient, but relatively few samples.

Fig. 2 shows the results for the considered small motion, also comparing the influence of different δ values. It is apparent that the small motion is more sensitive to the δ parameter. While for the large motion $\delta = 5$ already approximated the settling time of the linear penalty case using a few samples in the prediction horizon for each considered sampling frequency, as the sampling frequency increases, it is visible that a smaller δ value is needed to reduce the effect of N_control in case of the pseudo-Huber norm.

The explanation for this is twofold. Firstly, as the same N_control value means a shorter horizon for a larger f_s, it was anticipated that a small N_control would result in poorer performance at larger sampling frequencies. This is more distinguishable in case of the short motion because the ratio of the prediction window size and the motion time changes drastically with increasing N_control, furthermore, as the velocity constraint is inactive, the scaling between the velocities and δ is different. Secondly, δ has to be smaller for the second scenario because the same δ value for a large and small motion would mean a different percentage of motion time spent in the linear penalty range.

B. Iteration count

For the number of solver iterations, the value was logged after each OCP solution from the beginning of the simulation until settling was reached. For each objective function,

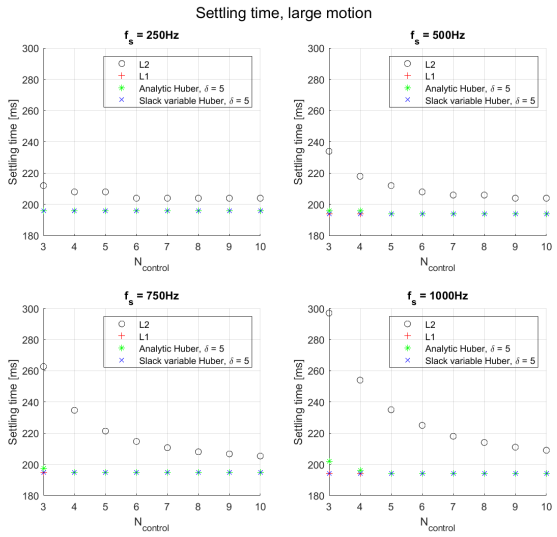


Fig. 1. Effect of different velocity penalty norms on the settling time considering various f_s and N_control values, large motion

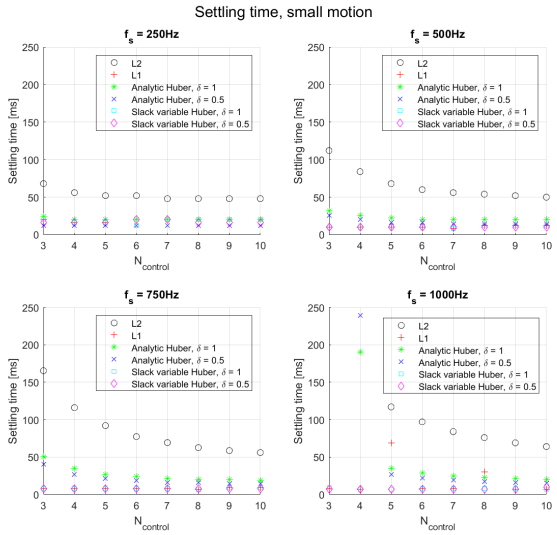


Fig. 2. Effect of different velocity penalty norms on the settling time considering various f_s and N_control values, small motion

N_control, and f_s, the average, the maximum value and the standard deviation were extracted.

For the large motion it was found that depending on N_control, the maximum number of iterations vary for each objective function and sampling rate in the range of 3–6 iterations; the change with respect to f_s is non-monotonously $\pm$1–2 iterations. Table I shows the average of all the average values and the maximum value for each objective function, while Fig. 3 shows the standard deviations of all 32 N_control–f_s combinations for each objective function.

The average number of iterations to solve an OCP with the pseudo-Huber velocity penalty is close to that of the quadratic penalty case, while using the linear penalty or the slack variable implementation of the Huber-norm is computationally more expensive due to the additional constraint(s).

TABLE I
STATISTICS OF THE NUMBER OF ITERATIONS – LARGE MOTION

$\dot{q}$ penalty	Quad.	Lin.	pseudo-Huber, $\delta = 5$	Huber, $\delta = 5$
mean	20.7	29.4	19.8	32.9
max	29	36	27	41

TABLE II
STATISTICS OF THE NUMBER OF ITERATIONS – SMALL MOTION

$\dot{q}$ penalty	Quad.	Lin.	pseudo-Huber, $\delta = 1$	pseudo-Huber, $\delta = 0.5$	Huber, $\delta = 1$	Huber, $\delta = 0.5$
mean	15.6	24.1	16.4	17.6	27.7	26.9
max	20	33	41	53	41	34

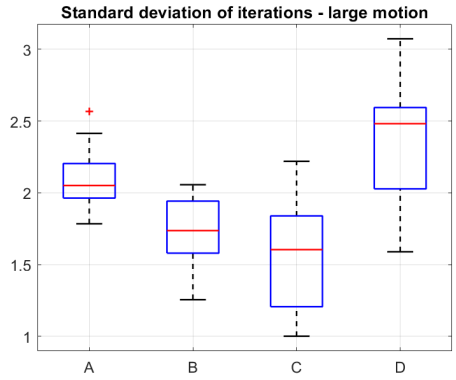


Fig. 3. Standard deviation of number of iterations for the large motion; A: L2, B: L1, C: pseudo-Huber, $\delta = 5$, D: Huber, $\delta = 5$

Table II shows the mean and worst case iteration values, and Fig. 4 shows the standard deviations for each objective function in case of the small motion. Using the quadratic penalty, neither N_control nor f_s affected the results; using the linear penalty there are differences, but there is no monotonous pattern with respect to N_control or f_s. Using the Huber-norm, as δ decreases, there are more and more outliers, but for different thresholds different N_control–f_s combinations produce these. The increasing standard deviation is partly due to the low number of samples comprising the motion. While the slack variable version of the Huber-norm increases computational complexity, due to its formulation as opposed to the nonlinear, analytic approximation the deviation is smaller and the iteration count contains less outliers.

Comparing the average and maximum number of iterations using the Huber-norm with using the linear penalty, it can be seen that the pseudo-Huber norm becomes computationally

more expensive as its smoothness decreases. The merit of the Huber-norm over the ℓ_1-norm in case of the small motion is rather the elimination of the overshoots that might occur.

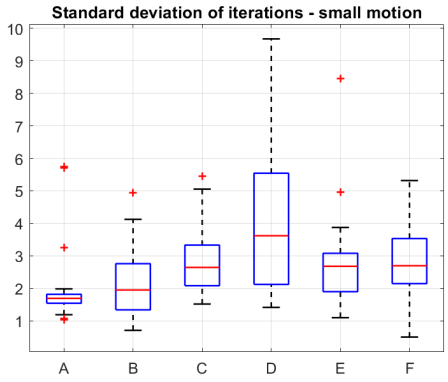


Fig. 4. Standard deviation of number of iterations for the small motion; A: L2, B: L1, C: pseudo-Huber, $\delta = 1$, D: pseudo-Huber, $\delta = 0.5$, E: Huber, $\delta = 1$, F: Huber, $\delta = 0.5$

C. Simulations with perturbed inertia parameters

In order to introduce noise – as SCARA robots are usually used for pick-and-place operations, and the model might not be accurate – 100 models were generated in which all inertia parameters of the joints were perturbed by $-10\% \ldots + 10\%$ using a uniform distribution.

The simulations have shown that depending the motion scenario the settling time and the statistics of the number of iterations per control action follow the same tendency as in the noise-free case with respect to N_{control}, f_{s} and the objective function, but with a magnified effect:

- in case of the large motion, the ℓ_1-norm and the slack variable version of the Huber-norm have higher computational complexity than the pseudo-Huber norm while achieving the same settling time;
- in case of the small motion, the ℓ_1-norm is unreliable for minimizing settling time while if δ is too small, the pseudo-Huber norm is erratic in terms of iteration count.

IV. Conclusions and Future Work

This case study presented via simulation results that by penalizing the velocity of the joints of a SCARA robot using a Huber cost function with a well-chosen threshold value, the time of a point-to-point motion can be decreased given a certain prediction horizon, or conversely, the size of the prediction horizon may be decreased and the motion is still approximately time-optimal. In case of a fast-reacting system this may be essential, as a relatively large horizon encompasses a larger number of control intervals as the sampling frequency increases, causing a stricter time-constraint on the computation time of each control action.

For large motions which exploit the velocity range of the robot, the pseudo-Huber norm is the best choice to mimic time-optimality with a low computational complexity, as δ can be chosen low enough to minimize time while high enough to ensure smoothness. For small, precision motions, much smaller δ values increase complexity and the deviation of the required number of iterations; however, the slack variable implementation of the Huber-norm has a consistently high computational cost. The best decision would be to compromise by using the pseudo-Huber norm with such a threshold so that time-optimality is sufficiently approximated (but not minimal), while it is high enough so that the standard deviation iteration count is low.

There are a number of avenues of ongoing or planned work to extend this study. The addition of process noise might be worthwhile, as the ℓ_1-norm and the Huber-norm with a small threshold may introduce deadbeat behavior which is sensitive to these disturbances. Another useful aspect would be to execute the tests on different plants and verify them experimentally.

Finally, for a realistic application either the solver should be accommodated to the requirements or the model should be simplified – if possible – to actually reach lower computation times than the sampling interval. With the currently used interior-point optimizer in a MATLAB environment none of the logged computation times are actually feasible for online control. Using a sequential quadratic program instead of the complete nonlinear program should ameliorate the results.

Acknowledgment

This research is supported by Flanders Make through ICON project ID2CON: Integrated IDentification for CONtrol.

References

[1] J. Rawlings, D. Mayne, and M. Diehl, Model Predictive Control: Theory, Computation and Design, 2nd Edition, Nob Hill Publishing, 2019.
[2] S. Boyd, and L. Vandenberghe, Convex Optimization, Cambridge University Press, 2004.
[3] M. Kiehl, Parallel multiple shooting for the solution of initial value problems, Parallel Computing, Volume 20, Issue 3, 1994, pp. 275-295.
[4] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. Mansard, The Pinocchio C++ Library: A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives, 2019 IEEE/SICE International Symposium on System Integration (SII), IEEE, 2019, pp. 614-619.
[5] A. Astudillo, J. Gillis, W. Decré, G. Pipeleers, and J. Swevers, Towards an Open Toolchain for Fast Nonlinear MPC for Serial Robots, IFAC-PapersOnLine, Elsevier, Volume 53, Issue 2, 2020, pp. 9814-9819.
[6] A. Astudillo, J. Carpentier, J. Gillis, G. Pipeleers, and J. Swevers, Mixed Use of Analytical Derivatives and Algorithmic Differentiation for NMPC of Robot Manipulators, Modeling, Estimation and Control Conference, Austin, Texas, 2021., in press
[7] J. Andersson, J. Gillis, G. Horn, J. Rawlings, and M. Diehl, CasADi: a software framework for nonlinear optimization and optimal control, Mathematical Programming Computation, Volume 11, Issue 1, Springer, 2019, pp. 1-36.
[8] J. Gillis, B. Vandewal, G. Pipeleers, and J. Swevers, Effortless modeling of optimal control problems with rockit, 39th Benelux Meeting on Systems and Control, Elspeet, The Netherlands, 2020.
[9] A. Wächter, and L. Biegler, On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming, Mathematical Programming 106, 2006, pp. 25-57.
[10] A. Dötlinger and R. M. Kennel, Near time-optimal model predictive control using an L1-norm based cost functional, 2014 IEEE Energy Conversion Congress and Exposition (ECCE), 2014, pp. 3504-3511.
[11] M. Fehér, O. Straka, and V. Šmídl, Model predictive control of electric drive system with L1-norm, European Journal of Control, Volume 56, 2020, pp. 242-253.

Bayesian analysis of multi-target genetic markers using hierarchical phenotypic data

Tamás Nagy*†, Nóra Eszlári†, Gabriella Juhász† Péter Antal*
*Budapest University of Technology and Economics, Department of Measurement and Information Systems
Budapest, Hungary
Email: {nagy.tamas, antal}@mit.bme.hu
†Semmelweis University, Department of Pharmacodynamics
Budapest, Hungary
Email: nagy.tamas@phd.semmelweis.hu, {eszlari.nora, juhasz.gabriella}@pharma.semmelweis-univ.hu

***Abstract*—Hierarchical data is ubiquitous in healthcare, but taking advantage of hierarchic information is still an open problem. We demonstrate the strengths and weaknesses of Bayesian multilevel analysis (BMLA) in this scenario by characterizing the multi-target relationships between genetic and phenotypic variables. The BMLA method does not scale well with the number of variables, thus we performed filtering using standard pairwise genome-wide association analysis. In this first GWAS phase, we selected only the most significant genotypic variables for further screening. We worked with various thresholds, resulting in 274, 12, and 2 genetic variables, these were treated as interventional variables in the second phase of data analysis using BMLA. The hierarchy of the phenotypic variables was given a priori, thus, we could estimate the posteriors of the relevance, i.e., for direct, non-mediated interaction between these broader categories and genetic markers. Additionally, we could approximate the joint relevance of genetic markers for multiple phenotypic groups, such as for mental health, metabolic and cardiovascular descriptors. The existence of such multi-target (pleiotropic) genetic factors is already indicated by significant genetic correlation between disease groups, i.e., by the overlap of their genetic background. Using this two-stage Bayesian systems-based method, we could robustly induce posteriors for these multivariate and multi-target relevance relations. The systematic investigation of a varying number of genetic and phenotypic factors confirmed that the method is sensitive enough to highlight the weak effects of genetic variables that can be easily overshadowed by the strong phenotypic correlations.**

***Index Terms*—bayesian statistics, multitarget, hierarchy**

I. Introduction

For us, humans it is very natural to order and organize data into hierarchical structures to gain more insight and have better maintenance capabilities, but we rarely use these structures computationally during advanced scientific research. The main reason behind this maybe the lack of methods supporting hierarchical data. This kind of data is abundant in healthcare related scenarios, e.g. the International Classification of Diseases (ICD) uses hierarchical categories, various scores use weighted sum of variables, which also defines a two-level hierarchy and genetic data also has multiple levels (single nucleotide polymorphisms (SNP), genes, gene sets, chromosomes).

A good example for hierarchical data processing is a novel Bayesian method called TreeWAS , which shows about 20% increase in statistical power compared to 'classical' methods, by utilizing the hierarchy of disease classification [1]. An other, a more general, regression like method, is the Hierarchical Linear Modeling (HLM) [2].

We worked on a Bayesian method that can be used in conjunction with classical GWAS tools and extend their capabilities. Our goal was to solve a Feature Subset Selection (FSS) problem, by finding hierarchic, multi-target, directly relevant genetic markers. Hierarchic, meaning a marker that is directly relevant to the target variables, on any abstraction level, and multi-target, meaning the marker can be relevant to multiple variables, or groups of variables. This work will be extended with additional post-processing and visualisation steps.

II. Methods

A. Data

Our test was based on the 'Budakalasz Health Examination Survey' dataset, which contains the necessary medical and genetic information for 629 participants, after excluding genetically related individuals. Our goals were to characterize the relationship between psychiatric and cardiovascular variables with respect to the SNPs found in the Catenin Alpha 2 gene (CTNNA2), which encodes a protein related to neural growth [13].

The cardiovascular status of the participants was assessed with the Framingham Risk Score [14]. This category included 4 scoring variables: FramCVD - cardiovascular disease, FramCHD - coronary hearth disease, FramHCHD - hearth diseases related to high HDL cholesterol, FramSTROKE - stroke.

For psychiatric assessment, the Brief Symptom Inventory (BSI) was used, which includes 9 sub-scales that were used on the variable level.

To characterize rumination, the Ruminative Response Scale (RRS) was used [4] with 3 parameters on the variable level.

B. Variable Selection with Linear Mixed Models Based Genome Wide Association Study

The samples were processed according to the basic GWAS quality control steps [8] and a linkage disequilibrium filtering was performed on the CTNNA2 region.

DOI 10.3311/MINISY2022-004

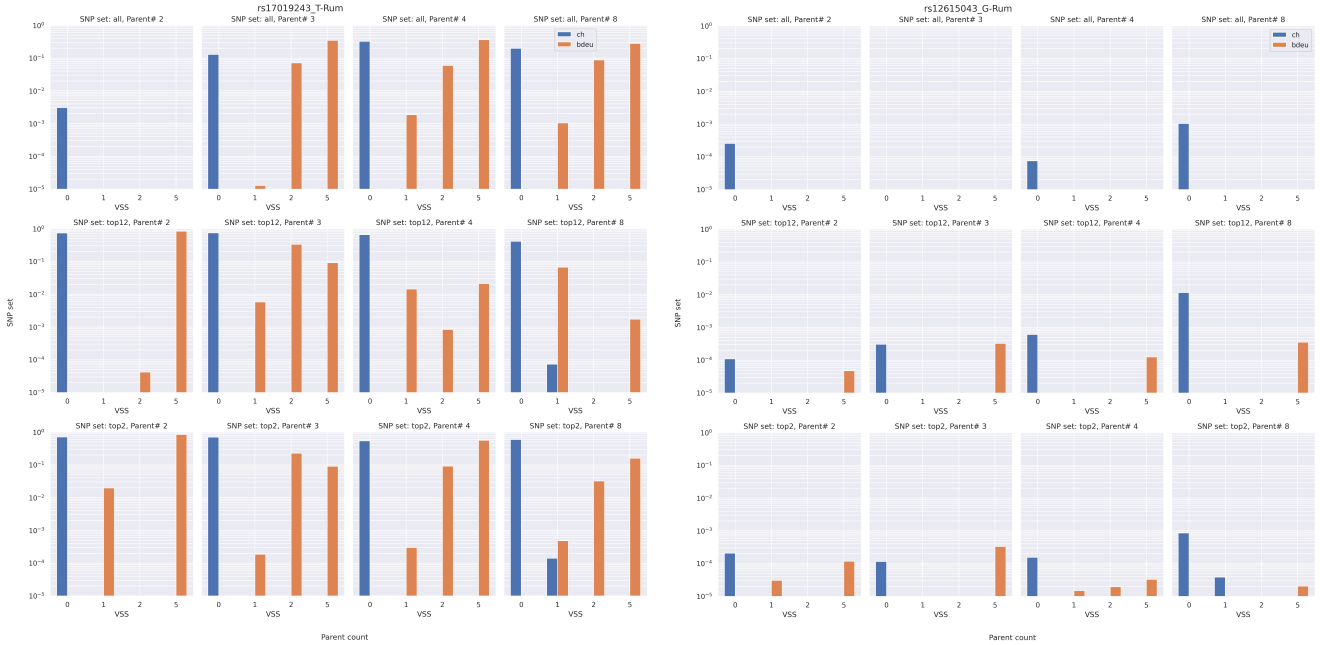


Fig. 1. Posterior probabilities between the rs17019243 SNP - RRS (on the left) and the rs12615043 SNP - RRS (on the right) based on a parameter sweep. The sweep was completed for various virtual sample sizes (VSS), maximum parent number per nodes, parameter priors (Bayesian–Dirichlet equivalent uniform - bdeu, Cooper-Herskovits - ch) and the number of included SNPs were gradually restricted based on relevance from the GWAS step. The number of burn-in (1 000 000) and sampling steps (5 000 000) were constant along the batches.

After these quality control steps, only 274 CTNNA2 related SNPs remained. This is still too much computational complexity for our hierarchical method to run calculations reliably, but enough restriction to do exploratory analysis. Thus, further narrowing of the parameter palette was necessary. This has been done by a Linear Mixed Models (LMM) based method, FaST-LMM [9], to determine the most significantly associated SNPs for each fenotype: the BMI, the Framingham components, the BSI components, and the RRS components. This yielded 12 distinct SNPs, from which the strongest 2 were selected for a further, auxiliary analysis.

C. Bayesian Multilevel Analysis

To better understand the directness of the relationships between the variables, we utilized the Bayesian network based Bayesian Multilevel Analysis (BMLA) [6]. The naturally rising two-level hierarchy was given explicitly was given explicitly to the model. An other option to utilize hierarchy would have benn the construction of mega-nodes that refer to higher level terms. This method is limited by the exponentional increase of the induced cardinality of the mega-node or the blurred representativity of the underlying dependency model. However, models at higher abstraction level could boost learning, so, we investigate the joint use of multiple models at varying abstraction levels, but this is out of scope for this study.

The Bayesian learning uses a Markov-Chain Monte Carlo based methodology, where we set 1 000 000 burn-in steps to guarantee MCMC convergence for pairwise features of variables in this cardinality range, followed by 5 000 000 actual sampling steps to approximate the most fitting directed acyclic graph (DAG) model for the data. Confidence of the estimates were estimated using multiple runs and we also performed systematic sensitivity analysis for changes in the variable set. Because the sample-size was low (n=629), we performed a sweep for various parameters of the BMLA tool.

We tested two parameter priors: the Cooper-Herskovits (CH) [5] and the Bayesian-Dirichlet equivalent uniform (BDeu) [10] [11]. And we also evaluated the effects of the virtual sample size, and the maximum number of parents per node.

To better grasp the effect of the number of genotipic variables, we conducted our measurements with various number of included SNPs, namely the original 274, the 12 with significant effect, and the 2 most significant SNPs.

The categorical interactions can be measured multiple ways. The first one is based on the probability of a direct edge between any of the variables of the category and the external variables, the second one is based on Markov Blanket Membership (MBM, see Fig. 3). In the present study we wanted to analyse the directness of rumination, thus the calculations were performed with the first option.

D. Visualisation

The output files from the BMLA were post-processed with a python script, then based on this the styling and visualization was done in Cytoscape v3.9.0. [12]. The post-processing included the separate labeling of edges between variable-variable level nodes and variable-category level nodes, the

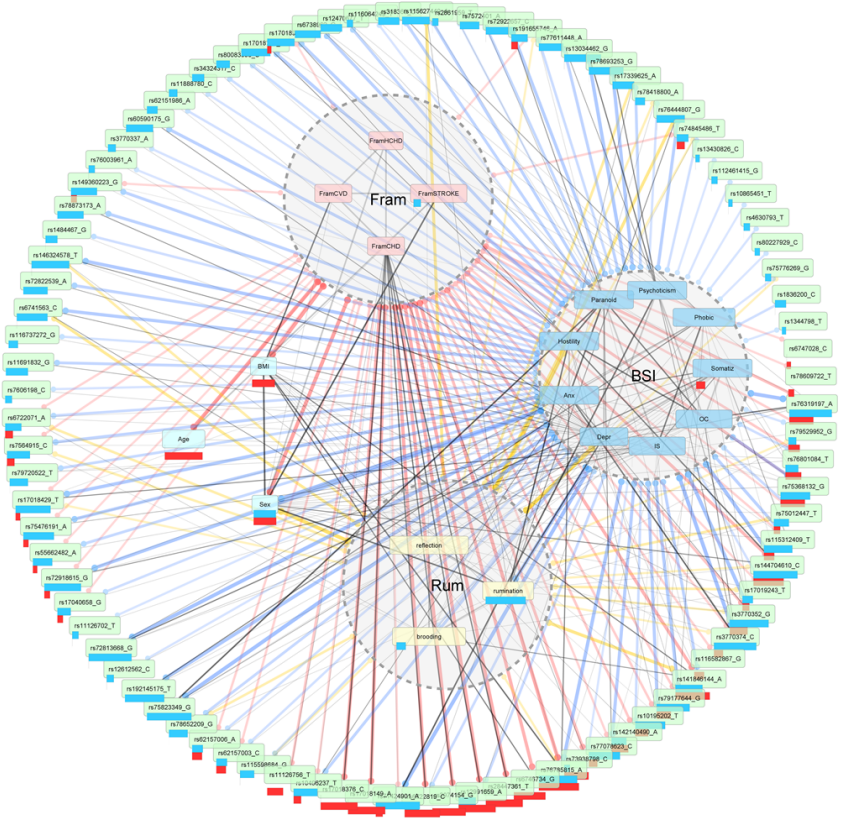


Fig. 2. The results of the BMLA including every SNP (for the purpose of visualisation, a limit of 0.1 for the posterior probability was introduced for edges and zero-degree nodes were hidden). The number of variables may make the interpretation of the figure challenging, but based on the hierarchic layout, one could easily determine SNP groups by relation with the main categories.

labeling of variable/category level nodes and the introduction of optional posterior limits for edge visibility.

III. Results

A multi-target BMLA was conducted with the 274 SNPs of the CTNNA2 gene, remaining after the GWAS step. The phenotpic variables were grouped based on broader categories, the sex, age and BMI variables, alongside the genotypic variables were set as exogenic variables. We ran calculations to determine the pleiotropic effects of the genetic variables and understand the directness of the variables. Because the BMLA method can be sensitive to parameter count, we iteratively reduced the number of genetic variables, leaving only the most significant ones, while performing a parameter sweep for every batch.

The results of the first batch, incorporating all the SNP variables (Fig. 2) shows that most of the SNPs have plausible relations (84 edges with posterior probability larger than 0.1, with the average of 0.47) to the BSI category, while the relation with the individual variables is not as pronounced, it is overshadowed by the stronger relations between the phenotipic variables. The Framingham and Rumination groups also have numerous connections. This batch shows that the method can detect the weak posteriors of the SNP related edges, but is not suitable for observations, because the limiting effects of the large variable pool on the BMLA method.

The second batch, containing only the 12 most relevant SNPs (according to the GWAS based selection step) can be seen on figure 4. Here we can see the two most relevant SNPs, rs17019243 and rs12615043, moderately standing out; these are propagated to the next batch.

After having all the SNPs sets defined we performed a parameter sweep, the results for the two most significant SNPs can be seen on figure 1.

IV. Discussion

We used GWAS related methods to reduce the number of variables in our multi-target study, where hierarchical data analysis was performed with BMLA.

Our study has some limitations, namely, the number of participants in the 'Budakalasz Health Examination Survey' dataset was low, thus the actual number of given rare poly-

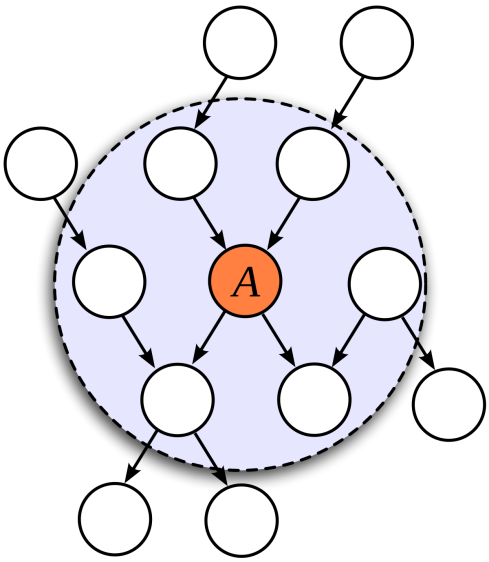


Fig. 3. Markov Blanket Membership: Node A's Markov blanket contains every direct parent of A, every child of A, and every child's parent. A posterior MBM probability is the probability that a node is contained inside the Markov blanket. [7]

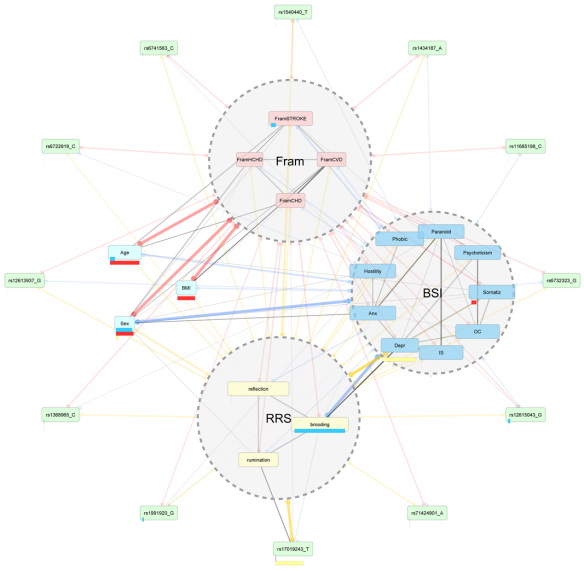


Fig. 4. The results of the BMLA including the top 12 most significant SNPs from the GWAS step. The two most significant SNPs (rs17019243 and rs12615043) can be seen on the bottom.

morphisms related to the CTNNA2 gene, was in the 10 to 100 magnitude, which is in fact the lower effective boundary of the BMLA method.

In the future this tool-set will be extended with an interactive visualisation tool which will aid exploratory analysis by providing ways to dynamically show associations across different levels in the hierarchy.

V. Acknowledgement

The research presented in this paper was supported by the Ministry of Innovation and the National Research, Development and Innovation Office within the framework of the Artificial Intelligence National Laboratory Programme.

Supported by the ÚNKP-21-3 New National Excellence Program of the Ministry of Human Capacities.

References

[1] Cortes, A., Dendrou, C. A., Motyer, A., Jostins, L., Vukcevic, D., Dilthey, A., … McVean, G. (2017). Bayesian analysis of genetic association across tree-structured routine healthcare data in the UK Biobank. Nature Genetics, 49(9), 1311–1318. doi:10.1038/ng.3926

[2] Woltman, H., Feldstain, A., Mackay, J. C., Rocchi, M. (2012) An introduction to hierarchical linear modeling, Tutorials in Quantitative Methods for Psychology, 8(1), 52-69. doi: 10.20982/tqmp.08.1.p052

[3] Derogatis, L. R., Melisaratos, N. (1983). The Brief Symptom Inventory: An introductory report. Psychological Medicine, 13(3), 595–605. https://doi.org/10.1017/S0033291700048017

[4] Treynor W., Gonzalez R., Nolen-Hoeksema S. Rumination reconsidered: A psychometric analysis. Cognit. Ther. Res. 2003;27:247–259. doi: 10.1023/A:1023910315561

[5] Cooper, G.F., Herskovits, E. A Bayesian method for the induction of probabilistic networks from data. Mach Learn 9, 309–347 (1992). https://doi.org/10.1007/BF00994110

[6] Leeuw, J. de, Meijer, E. (Eds.). (2008). Handbook of Multilevel Analysis. doi:10.1007/978-0-387-73186-5

[7] https://upload.wikimedia.org/wikipedia/commons/thumb/e/eb/Diagram_of_a_Markov_blanket.svg/1200px-Diagram_of_a_Markov_blanket.svg.png

[8] Quality control, imputation and analysis of genome-wide genotyping data from the Illumina HumanCoreExome microarray. Coleman JR, Euesden J, Patel H, Folarin AA, Newhouse S, Breen G; Brief Funct Genomics. 2016 Jul; 15(4):298-304.

[9] Lippert, C., Listgarten, J., Liu, Y. et al. FaST linear mixed models for genome-wide association studies. Nat Methods 8, 833–835 (2011). https://doi.org/10.1038/nmeth.1681

[10] Buntine WL, Theory refinement of Bayesian networks., In: Proc. of the 7th Conf. on Uncertainty in Artificial Intelligence (UAI-1991), Morgan Kaufmann, 1991, pp. 52–60, DOI 10.1.1.52.1068.

[11] Heckerman D, Geiger D, Likelihoods and parameter priors for Bayesian networks, MicroSoft Research., 1995. Tech. Rep. MSR-TR-95-54.

[12] Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T. Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome Res, 13:11 (2498-504). 2003 Nov. PubMed ID: 14597658.

[13] Schaffer AE, Breuss MW, Caglayan AO, et al. Biallelic loss of human CTNNA2, encoding αN-catenin, leads to ARP2/3 complex overactivity and disordered cortical neuronal migration. Nat Genet. 2018;50(8):1093-1101. doi:10.1038/s41588-018-0166-0

[14] Ralph B. D, Agostino, Sr, Ramachandran S. Vasan, Michael J. Pencina, Philip A. Wolf, Mark Cobain, Joseph M. Massaro and William B. Kannel. General cardiovascular risk profile for use in primary care: the Framingham Heart Study. Circulation 2008 February 12, 117 (6): 743-53

Dependability Modeling of Cyber-Physical Systems in the Gamma Framework

Richárd Szabó, András Vörös
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: richard.szabo@edu.bme.hu, voros.andras@vik.bme.hu

***Abstract*—Cyber-physical systems (CPS) can be found everywhere: smart homes, autonomous vehicles, aircrafts, healthcare, agriculture, and industrial production lines. CPSs are often critical, as system failure can cause serious damage to property and human lives.**

Today's cyber-physical systems are extremely complex, heterogeneous systems, so rigorous engineering approaches are needed both at design and runtime. On one hand, model-based techniques support the efficient system design, and on the other hand, fault-tolerant middleware and communication technologies support the reliable operation of critical CPS. However, modeling dependability-related system aspects is far from trivial. In this paper, our goal is to show a methodology that introduces design patterns for dependability modeling in the Gamma modeling framework to take a step towards the efficient design of dependable CPSs.

***Index Terms*—model-based development, cyber-physical system, CPS, formal analysis, dependability**

I. Introduction and Background

Cyber-physical systems (CPS) are complex, heterogeneous systems that contain a wide variety of devices, from sensor networks through edge devices to cloud-based services. CPSs often provide critical services, where the correct and reliable operation is crucial. Model-based techniques support not only the design of critical CPSs, but also formal verification techniques can be applied to system models to find design flaws and stochastic analysis techniques ensure that the system meets the extra-functional requirements. Ensuring dependable operation necessitates the application of fault-tolerant design/architecture patterns. However, model-based tools do not provide support to efficiently design complex dependability-related design solutions, where the services and also allocation is properly modeled.

In this paper, our goal is to make a step towards the design of dependable CPSs by proposing an approach that provides design support for engineers. Our proposed approach is model-driven and relies on an open-source design tool, the Gamma framework. Our long-term goal is to transform Gamma functional architecture models to system models enriched with the chosen design patterns of the services. In addition, our goal is to operate the services by using technologies with built-in dependability mechanisms. In the following, the background of our work is summarized.

This work was partially funded by the EC and NKFIH through the Arrowhead Tools project (EU grant No. 826452, NKFIH grant 2019-2.1.3-NEMZ ECSEL-2019-00003).

A. Dependability

Dependability is the ability of a system to avoid service failures that are more frequent and more severe than is acceptable [10].

Our approach focuses on the following aspects of dependability:

- **availability**: readiness for correct service.
- **reliability**: continuity of correct service.
- **safety**: absence of catastrophic consequences on the user(s) and the environment.

One of the means to ensure the dependability of the system is fault tolerance, which is the ability of the system to avoid service failures in the presence of faults. Our approach applies the following fault tolerance techniques to ensure the dependability of the system:

- **error detection**: identification of the presence of an error.
- **fault masking**: systematic usage of compensation to conceal a possibly progressive failure.

B. Model-driven Development

Model-driven development is a system design approach where the main product of development is the models that describe the system. These models contain valuable information about our system. Using the designed models, we can derive additional models and generate software source code/configurations. By using properly certified code generators, we can ensure the quality of the generated code, thus avoiding errors from human coding, and ensuring the quality of critical systems. We can also derive analysis models from the engineering models.

C. Formal Verification and Dependability Analysis

In our approach, we rely on models as the main artifact of the development. From the developed models we can derive formal/analysis models and run qualitative analysis to find causes of potential hazards and the effects of faults, and quantitative analysis to calculate reliability and availability of the system. Formal verification is a technique to exhaustively examine the behaviour of the systems and prove the error-free behaviour. Dependability analysis is based on stochastic

models of the system: traditional engineering models have to be enriched with fault-related probabilities and environmental assumptions and stochastic analysis is used to ensure dependability measures of the design of the system under the given environmental conditions.

D. Gamma Statechart Composition Framework

Gamma is a statechart composition framework [1] designed to model and verify component-based reactive systems and generate code from the models. The tool supports the hierarchical composition of the statecharts, which enables engineers to focus on subcomponents of the complex systems. The framework raises a new modeling layer over engineering state machines to describe communication between components, making it suitable for modeling complex, hierarchical systems. The tool allows the user to formally verify the models using UPPAAL [2] and Theta [11]. Finally, source code can be generated from the models.

E. Related Work

Various approaches exist in the literature to design dependable systems. The most commonly used architecture design techniques in practice are manual construction of models and using automatic techniques to synthesize full systems from components. During the manual construction, the engineer has to build everything from the ground up using well-defined modeling languages like SysML [5], but has the freedom to use any components, technologies and algorithms, at the cost of having to implement them. Automated techniques on the other hand like ArcheOpterix [7], can provide optimized architecture, but at the cost of only building it from a set of predefined set of elements.

In [6] a UML profile is defined to aid the dependability analysis of real-time systems. The DAMRTS (Dependability Analysis Models for Real-Time Systems) profile supports the modelling of the probabilistic aspects of systems defines a transformation of UML models to probabilistic timed automata.

A more comprehensive summary of the tools and algorithms can be found in [12].

II. Overview of the Approach

This section details our approach for the dependable design of component-based systems. We identified the following steps that should be supported by the Gamma tool and the planned transformations.

1) **Functional architecture**: The engineer designs the functions/services of the system using the Gamma Statechart Composition Framework. In this step, the focus is on the intended functions/functionalities. The output of this step is the function definitions as statecharts and the functional architecture.
2) **Environment modeling**: In the second step, the engineer enriches the model with the environment information. The environment is described by stochastic events or in the case of a complex environment, stochastic behaviour models provide the inputs (and faults) for the functional model. The environment can include various inputs from the world outside of the system, possible failure causes for both hardware and software components. E.g. hardware failures can be caused by overheating from a heatsource outside of the system.
3) **FMEDA**: In the third step, using the functional model and the environment model the engineer manually performs the *Failure modes, effects, and diagnostic analysis* (FMEDA) to determine the error propagation and compute the effects. The output of the analysis shows which components of the system require fault-tolerant patterns to meet the extra-functional requirements.
4) **Applying dependabilty design patterns**: In the fourth step, the engineer can apply design patterns based on the output of the FMEDA. Our goal is to support this step: we plan to provide a library of fault-tolerant patterns and corresponding transformations. At first, according to the FMEDA, the engineers have to choose the design pattern to be applied on the (critical) functions. The tool will transform the model and apply the design pattern. In addition, when the design pattern has to be fine-tuned for the domain, the engineer can modify the result accordingly by modifying/redefining the logic itself. Our idea is not only to transform the model, but also to define the constraints for deployment as annotations on the model. Engineers can also configure the deployment by changing the annotations.
5) **Verification and extra-functional analysis**: In the fifth step, the verification and analysis can be performed. Formal verification can be applied in two phases: at first, the functional architecture model has to be verified for providing the intended functionalities. Then, the dependable functionalities and error propagation in the system can also be verified by the advanced techniques provided by Gamma. The dependability model of the system containing the environmental and fault information is analyzed by the stochastic analysis algorithms in the stochastic Gamma [9]. When deployment is also provided, the whole system can be evaluated from the dependability point of view.

After the final step if the results of the analysis is not acceptable the process can be repeated from Step 4. or a total redesign of the system is needed.

III. Desigh Pattern Library for Dependability

During the design of the system, architectural design patterns can be used to ensure the dependability of the system. The following design patterns were chosen to apply the fault tolerant techniques and aspects presented in Section I-A.

Our plan is to provide a library supporting some of the well-known design patterns for error detection and fault-tolerant behaviour [4]. In order to reach the target dependability measures, the applied design patterns must also follow deployment constraints such as that variants must be deployed on separate devices with similar capabilities. In addition, different

technologies can be used to increase the fault tolerance of the system, but those can add additional constraints during the system design (e.g. Kubernetes [3] works with distributed systems, thus only asynchronous components can be used in the system).

In the following, each pattern is presented on an example seen in Fig. 1. The engineer designs a part of the system which contains three components connected to each other. The FMEDA analysis shows that the component in the middle (*Component_2*) is a single-point-of-failure, so the engineer decides to use a pattern to increase the fault tolerance of the system.

Fig. 1. Setup for the examples.

The fault-tolerant patterns provided by the library are the following:

Category	Fault-tolerant pattern
Error detection	Two-channel architecture with comparison
Error detection	Two-channel architecture with safety checking
Fault tolerance (HW)	N-modular redundancy
Fault tolerance (SW)	N-version programming
Fault tolerance (SW)	Recovery blocks

A. Error Detection Patterns

1) Two-channel Architecture with Comparison: The two channels work on shared input, with comparison of the outputs. This pattern provides high error detection coverage, but can have increased detection latency.

- **Parameters**: Tolerance for accepting outputs as equal.
- **Constraints**: The channels must be deployed separately to avoid common mode faults of the hardware components, which could cause the faulty behaviour of both channels.

In this example the engineer annotates *Component_2* with *Two-channel Architecture with Comparison (2CC)* from the library as seen in Fig. 2. After the annotation, model transformation generates the components defined in the pattern. The 2CC pattern defines a second channel of the annotated software component, a *DataMultiplier* to share the data between the channels and a *Comparator* to compare the data from the channels and raise an error signal if the output is outside the acceptance range.

2) Two-channel Architecture with Safety Checking: This pattern provides an independent channel for safety checking.

- **Parameters**: Explicit safety rules.
- **Constraints**: -

In this example the engineer annotates *Component_2* with *Two-channel Architecture with Safety Checking (2CS)* from the library as seen in Fig. 3. After the annotation, model transformation generates the components defined in the pattern. The

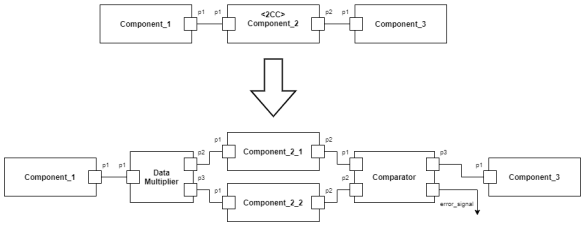


Fig. 2. Model transformation for Two-channel Architecture with Comparison.

2CS pattern defines a safety channel to the annotated software component, a *DataMultiplier* to share the data between the channels and a *Safety Checker* to check the data from the channels and raise an error signal if the output is outside the acceptance rules. The generated safety channel is only a skeleton, the safety engineer must implement the logic of the channel according to the parameters.

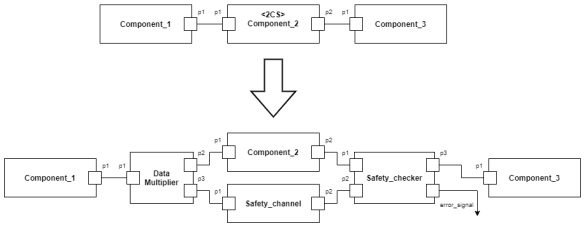


Fig. 3. Model transformation for Two-channel Architecture with Safety Checking.

B. Fault tolerance Patterns for Permanent Hardware Failure

1) N-modular Redundancy: This pattern provides fault tolerance by masking the failure with majority voting.

- **Parameters**: Number of modules.
- **Constraints**: The modules must be deployed separately to avoid common mode faults.

In this example the engineer annotates *Component_2* with *N- Modular Redundancy (NMR)* from the library similarly as seen in Fig. 4. After the annotation, model transformation generates the components defined in the pattern. The NMR pattern defines the replication of the annotated component N times (where N is the parameter given by the engineer), a *DataMultiplier* to share the data between the replicas and a *Voter* to collect the data from the replicas and decide the output with majority vote. The logic of the generated *Voter* can be modified to achieve more complex fault tolerance.

C. Fault tolerance Patterns for Software Failure

1) N-version Programming: This pattern provides active redundancy by using multiple software modules with diverse implementations, algoritms and programming languages.

- **Parameters**: Number of variants, explicit acceptance rules.
- **Constraints**: The modules must be deployed separately to avoid common mode faults.

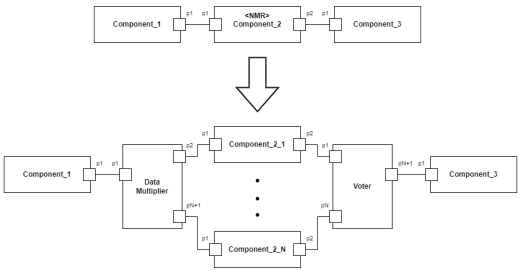


Fig. 4. Model transformation for N-Modular Redundancy.

In this example, the engineer annotates *SW_Component_2* with *N-Version Programming (NVP)* from the library as seen in Fig. 5. After the annotation, model transformation generates the components defined in the pattern. The NVP pattern defines the replication of the annotated software component N times (where N is the parameter given by the engineer), a *DataMultiplier* to share the data between the replicas and a *Voter* to collect the data from the replicas and decide the output with majority vote and to provide error signal if the output is outside the acceptance range. The logic of the generated *Voter* can be modified to achieve more complex fault tolerance.

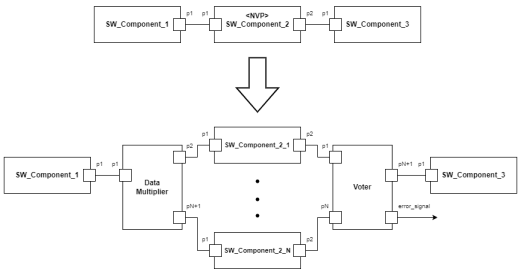


Fig. 5. Model transformation for N-Version Programming.

2) Recovery Blocks: This pattern provides passive redundancy with multiple software modules and acceptance checking. The next module is executed only if the previous fails on the acceptance check.

- **Parameters**: Number of modules, explicit acceptance rules.
- **Constraints**: -

In this example, the engineer annotates *SW_Component_2* with *Recovery Blocks (RB)* from the library as seen in Fig. 6. After the annotation, model transformation generates the components defined in the pattern. The RB pattern defines the replication of the annotated software component N times and *Checker* N times (where N is the parameter given by the engineer) in a chain, a *DataMultiplier* to share the data between the replicas. If the output of a replica is outside the acceptance range the next replica is executed, if there are no more replicas an error signal is sent.

IV. Conclusion

In this paper, we presented a design approach to design dependable CPSs and we defined a library of dependability-related design patterns to aid the design process. In the future, we plan to implement the approach and evaluate on industrial case studies. In addition, the formerly introduced deployment modeling approach [8] will also be integrated.

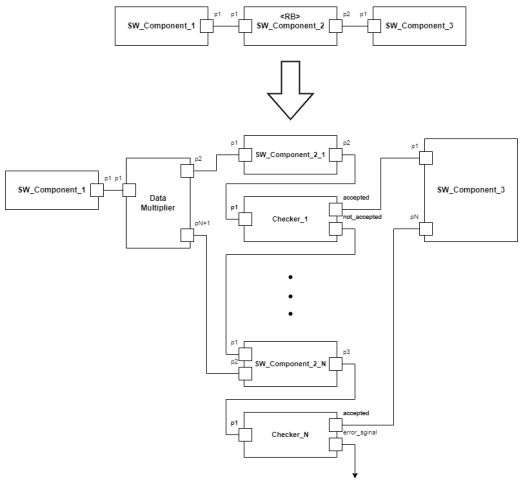


Fig. 6. Model transformation for Recovery Blocks.

References

[1] V. Molnár, B. Graics, A. Vörös, I. Majzik, and D. Varró, "The Gamma Statechart Composition Framework: design, verification and code generation for component-based reactive systems," in Proceedings of the 40th International Conference on Software Engineering: Companion Proceeedings, 2018, pp. 113–116. doi: 10.1145/3183440.3183489.

[2] Johan Bengtsson, Kim G. Larsen, Fredrik Larsson and Paul Pettersson, Wang Yi, "UPPAAL — a Tool Suite for Automatic Verification of Real–Time Systems". 1995.

[3] The Kubernetes Authors, "Kubernetes".https://kubernetes.io.

[4] István Majzik. "IT System Design – Safety-critical Systems". Budapest University of Technology and Economics, Department of Measurement and Information Systems. 2020.

[5] Object Management Group. "OMG Systems Modeling Language (OMG SysML)". Available at: http://www.omg.org/spec/SysML/.

[6] Nawal Addouche, Christian Antoine, Jacky Montmain. "UML Models for Dependability Analysis of Real-Time Systems". SMC04, IEEE International Conference on Systems, Man and Cybernetics, 2004, La Hague, Netherlands. ffhal-00354034f

[7] Aldeida Aleti, Stefan Björnander, Lars Grunske, Indika Meedeniya. (2009). "ArcheOpterix: An extendable tool for architecture optimization of AADL models". Proceedings of the 2009 ICSE Workshop on Model-Based Methodologies for Pervasive and Embedded Software, MOMPES 2009. 61-71. 10.1109/MOMPES.2009.5069138.

[8] Richárd Szabó, András Vörös. "Towards formally analyzed Cyber-Physical Systems". 17th European Dependable Computing Conference (EDCC 2021).

[9] Simon József Nagy, Bence Graics, Kristóf Marussy & András Vörös. "Simulation-based Safety Assessment of High-level Reliability Models". *4th Workshop On Models For Formal Analysis Of Real Systems*. (2020)

[10] Algirdas Avizienis, Jean-Claude Laprie, Brian Randell & Carl Landwehr. "Basic concepts and taxonomy of dependable and secure computing". *IEEE Transactions On Dependable And Secure Computing*. **1**, 11-33 (2004)

[11] Tamás Tóth, Ákos Hajdu, András Vörös, Zoltán Micskei & István Majzik. "Theta: a Framework for Abstraction Refinement-Based Model Checking". *Proceedings Of The 17th Conference On Formal Methods In Computer-Aided Design*. pp. 176-179 (2017)

[12] Bernardi, Simona and Merseguer, José and Petriu, Dorina C. "Dependability Modeling and Analysis of Software Systems Specified with UML" *ACM Comput. Surv.* Volume **45.** Issue **1** November 2012 Article No.: 2pp 1–48

Design of an Audio Frequency Range Distributed Data Acquisition System Prototype

András Wiesner
Department of Measurement and Information Systems
Budapest University of Technology and Economics
Budapest, Hungary
awiesner@edu.bme.hu

***Abstract*—Use of packet-based real-time audio and video transmission gets more and more common nowadays. These systems consist of precisely synchronized distributed nodes, the synchronization is usually done using the IEEE 1588 Precision Time Protocol. For example, the well-known Dante system also utilizes PTP as the synchronization solution, and Audio Video Bridging (IEEE 802.1BA-2011) as well. In this paper I introduce a prototype system designed for hardware-level sampling synchronization based on the STM32H743 480MHz Cortex-M7 microcontroller with the aim of describing the synchronization algorithm. A custom extension board has been also made tailored to the NUCLEO development board featuring the MCU to provide us with the required analog and the synchronized I2S-interface This board gives a place for the TLV320AIC23BPW stereo CODEC performing A/D and D/A conversions. This system is designed for audio-frequency range – that's why I use audio a CODEC instead of discrete A/D and D/A converters.**

***Index Terms*—IEEE1588 PTP, Audio Video Bridging, Networked embedded systems, Clock synchronization, Synchronized sampling, Performance evaluation**

I. Introduction

Nowadays demand on synchronized sampling in networked, distributed systems on distant nodes is getting higher and higher. Not only does professional equipment provide means of precise synchronization but lower-tier gears of broader range featuring such methods are more frequent as well. Precise time and clock synchronization, essential for such applications, may be done using several methods, but especially in networked system synchronization using IEEE 1588 Precision Time Protocol [1] (PTP) is quite handy: no need for extra wiring and extra network and also network equipment supporting PTP-message processing is getting more common. In professional audio processing and transmission, a tendency of digital tools and methods overtaking the analog ones is visible, this is especially true in the case of video transmission. Systems capable of such are the Audio Video Bridging systems, which could not operate without a seamless precise time and clock synchronization. The first Audio Video Bridging (AVB) systems were designed for professional use but nowadays we can notice the increasing usage by non-professionals too. For now Ethernet standards including specific PTP profiles have been established particularly for use in Ethernet-based AVB [2] systems, for example, the IEEE 802.1as [3] standard defining synchronization methods for use in Audio Video Bridging.

In this paper, I introduce the prototype I have designed using the STM32H743 [4] MCU, and I am going to mainly focus on the hardware level sampling synchronization. In the end I prove the system sampling accuracy through numerous measurements.

II. Specification

A. General Considerations for Distributed, Synchronized Data Collector Systems' Endpoints

I've found the following points the most important a synchronized endpoint should satisfy:

- The system should be capable of clock synchronization. Considering Ethernet-based data collector systems in most cases it is carried out using IEEE 1588 Precision Time Protocol.
- The system should feature hardware-level sampling synchronization. This is mostly done utilizing some sort of precise clock divider circuitry which allows tuning during operation.
- It should provide the users of a simple way of attaching to an already operating network, without causing any interruption.
- The system should be equipped with some kind of analog inputs and/or outputs where analog signals can enter or leave the device. Devices working in the audio frequency range mostly feature audio CODECs for managing A/D and D/A conversions, designed particularly for such applications.

B. Specification of the Prototype System

The prototype system has the following main aspects:

The system is based on the STMicroelectronics' STM32H743ZI [4] 480MHz ARM Cortex-M7 microcontroller. This device has an integrated Ethernet MAC supporting PTP hardware timestamping and it features multiple I^2S-interfaces interconnecting the MCU and the CODEC. A NUCLEO-H743 [5] development board is used as base hardware environment. The MCU on the board is clocked from a crystal oscillator I've installed on the board after I had found the board controller's clock output was unstable.

The CODEC by type is the TLV320AIC23BPW [6], manufactured by Texas Instruments. It features a stereo input and

DOI 10.3311/MINISY2022-006

output – respectively one stereo A/D and one stereo D/A converter – up to 96kHz data rate and 32-bit precision. Additional features include a microphone preamplifier and biasing circuit and a headphone amplifier to drive low impedance loads.

IEEE 1588 PTP is utilized for clock synchronization over Ethernet, and the CODEC clock signals are synchronized to the PTP-synchronized hardware clock. For the software stack, flexPTP [7] is used. CODEC clock is generated using one of the microcontroller's built-in PLLs.

The system's software architecture is built upon FreeRTOS and lwIP. The vendor-given network drivers have been modified to support PTP-timestamping.

The system features an automatic network discovery service helping the procedure of opening connection onto the device. Samples are transferred over Ethernet in streams of UDP packets using a non-standard, custom data structure.

An extension board has been designed to make it possible to connect the CODEC with the NUCLEO development board and to give place for the signal level analog circuitry. The board offers two exclusive inputs and two outputs: a mic and a line-level input and a line level and a low-impedance output for headphones. According to measurements design intents for noise coupling reductions were successful, the input circuitry only displays a noise approximately with 0.4mV standard deviation. The extension board analog circuitry

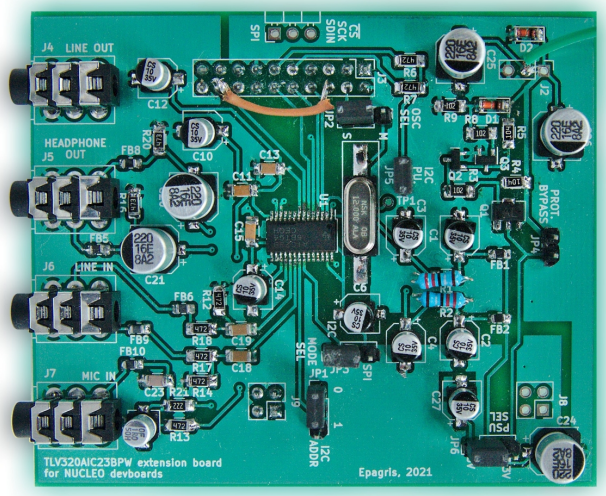


Fig. 1: Extension board top view

In the following, I'm going to focus on sampling synchronization, but I also describe the details mandatory for understanding sampling synchronization.

C. Time Synchronization Using IEEE 1588 Precision Time Protocol

The IEEE 1588 Precision Time Protocol (PTP) is a standard method of precise clock synchronization over Ethernet networks. PTP-networks consists of at least one master clock featuring some kind of an accurate time base, and of many slave-clock usually having less accurate timebases. The synchronization itself is done by running controller or servo algorithms receiving the momentarily time error in every synchronization cycle.

Hardware clocks (e.g. PTP clocks) often feature an nPPS (n Pulse-per-Second), most often a 1PPS signal output, which is a square wave signal of n Hz having a distinguished edge synchronized in phase to the hardware clock. (For example rising edges in this signal are tied to seconds rollover.) The nPPS signal along providing inter-device synchronization verification means, also enable external devices to synchronize to the hardware clock.

The software implementation I use for PTP synchronization is the flexPTP [7] ported to the current microcontroller. Clock increment is 5ns, tuning precision is 0.2328ppb. For tuning the frequency I use a simple PD controller.

D. I^2S-Protocol

The Inter Integrated Sound bus is a synchronous serial audio stream transmission bus designed by Phillips Semiconductors in 1986 [8]. The transmission is always point-to-point, usually with one master and one slave device. The transmission unit is a frame, containing 16-32-bit (per channel) full stereo samples.

Strictly the protocol defined three signals:

- a *Serial Clock (SCK)*, clocking the transmission,
- a *Word Select (WS) or Frame Select (FS)*, that selects the channel for being transferred and initiates transmission,
- a *Serial Data (SD)*, through which the actual sample data gets transcieved.

The transmission is unidirectional, to achieve full-duplex transmission, two synchronous I^2S-buses should be joined.

E. Master Clock Generation

Although, the signals mentioned above are sufficient for managing I^2S-transmissions but usually the CODEC needs a higher frequency Master Clock (MCLK) as well to drive the converters. The I^2S-signals SCK and FS derive from the MCLK generated on the master device, that's why they are synchronous to MCLK.

The MCLK is being generated using one of the MCU's built-in PLLs. In the system, the CODEC is configured to work with 16-bit samples using $f_s = 48$kHz sampling rate. According to the CODEC's datasheet, the MCLK frequency matching these settings is 12.288MHz. This frequency is generated by subsequent divisions of a higher (98.304MHz[1]) frequency signal generated by a PLL using the crystal oscillator as input source. The PLL offers a non-integer, fractional feedback divider which allows sufficiently precise output frequency tuning. However, it is significantly worse than frequency divider found in the PTP hardware clock; therefore, it is a future research direction how this part of the system can be improved.

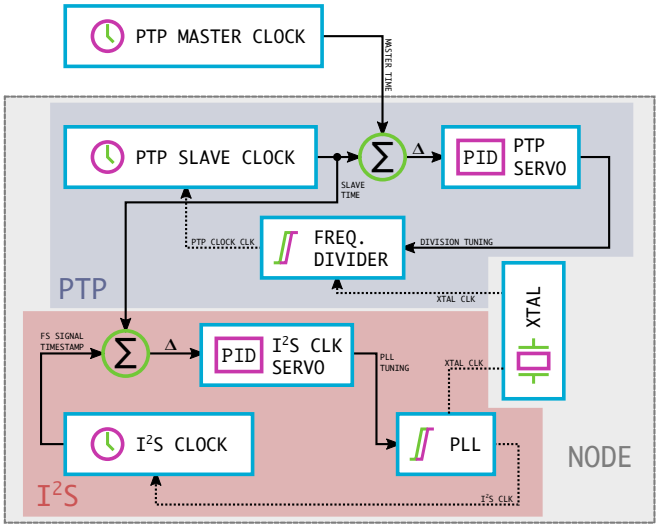


Fig. 2: Cascade control system formed by the PTP master-slave control path and PTP slave-$\mathtt{I^2S}$ path

F. PTP-$\mathtt{I^2S}$ Cascade Control

Sampling synchronization is achieved by synchronizing the sampling clock to the PTP master clock through the $\mathtt{I^2S}$ to local PTP slave clock to remote PTP master clock cascade control path depicted on fig. 2.

Control methods utilize tuneable frequency generating elements in both loops, a precision frequency divider in the PTP-loop and a PLL in the $\mathtt{I^2S}$ loop. Phase synchronization is performed through frequency tuning which avoids breaking time monotonicity. At startup, $\mathtt{I^2S}$-synchronization starts only after the PTP clock has settled.

PTP time error is calculated using the PTP's error calculation algorithm, in software (based on preciose hardware timestamps), $\mathtt{I^2S}$ time error is calculated using an advanced algorithm described in the next chapter.

G. Algorithm of the Frame Select Signal (FS) Synchronization

MCLK can not be directly synchronized to the PTP-controlled hardware clock. We achieve MCLK synchronization by synchronizing the FS signal to the 1PPS signal. According to the previous section, FS is synchronous to MCLK and vice versa, that's why synchronizing FS also synchronizes MCLK.

Time error between FS and the 1PPS edges can not be determined directly due to the difference in the frequencies of the signals. FS has a frequency equal to the sampling rate (48kHz), 1PPS's frequency is 1Hz.

The novel idea I introduced and applied here is the following: synchronize every Nth edge of the FS signal to the edges of a virtual, mathematically created reference k-PPS signal. N should be chosen so that $k = \frac{f_s}{N}$ is an integer. In other words: a one second long portion of the FS signal should be exactly split up into k pieces of N periods long portions.

[1]This frequency is 8 times greater than specified MCLK frequency. This is the recommended frequency for driving $\mathtt{I^2S}$-related hardware blocks, which manage further clock divisions.

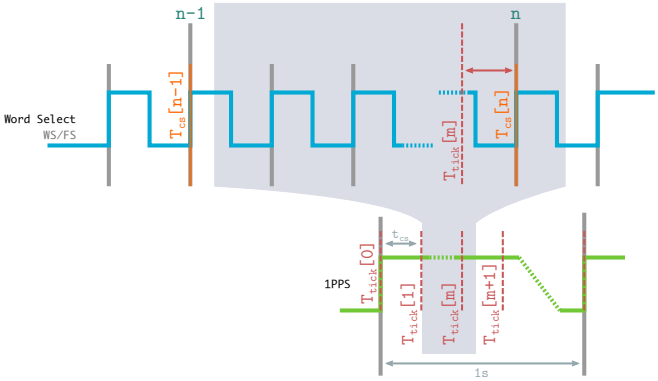


Fig. 3: FS signal synchronization using virtual k-PPS signal

Using MCU's special hardware features we can obtain timestamp to every N-edge of the FS signal. Taking advantage of k and N are integer as well, synchronization can be done only considering timestamps' sub-second fields. The reference k-PPS signal mth timestamp – not considering the seconds part – is mathematically calculated as follows:

$$T_{\text{tick}}[m] = m \cdot \underbrace{\frac{1}{k}}_{t_{\text{kPPS}}}, \; m \in [0..(k-1)|\mathbb{N}]. \tag{1}$$

For every FS timestamp T_{FS} we can always find a the closest k-PPS edge:

$$m_{\text{closest}} = \mathtt{round}(\frac{T_{\text{FS}}}{t_{\text{kPPS}}}) = \mathtt{round}(T_{\text{FS}} \cdot k). \tag{2}$$

Combining the previous equations time error (Δt) to this closest edge is as follows:

$$\boldsymbol{\Delta t = T_{\text{FS}} - \underbrace{\frac{\mathtt{round}(T_{\text{FS}} \cdot k)}{k}}_{m_{\text{closest}} \cdot \frac{1}{k}}} \tag{3}$$

Running an adequate servo algorithm tuning the PLL which produces MCLK, it can eliminate time error and synchronize FS to the PTP hardware clock.

In the system, I use a simple PD-controller for this purpose as well.

III. Performance Evaluation

A. Kinds of measurements

I carried out two types of measurements:

- I made long term measurements on PTP synchronization quality by logging time error printed on the node's software console. According to the physical signal time interval measurements, time error values reported by the software are accurate. This paper does not focus on PTP subsystem evaluation, but having a stable time synchronization is mandatory for synchronized sampling.
- I also made a long-term measurement on synchronization quality. I've been logging the FS signal edge-to virtual k-PPS signal edge time errors.

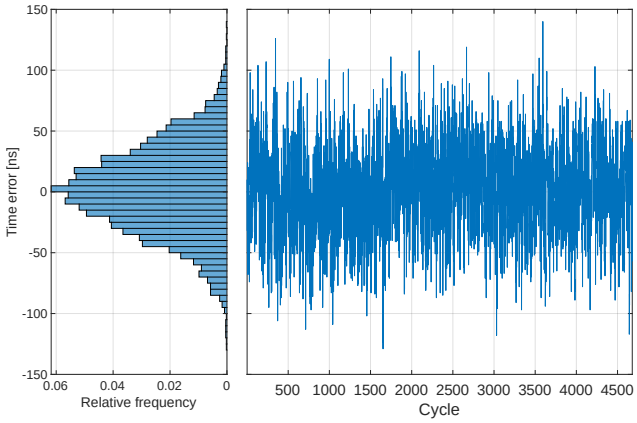


Fig. 4: Long term PTP stability measurement

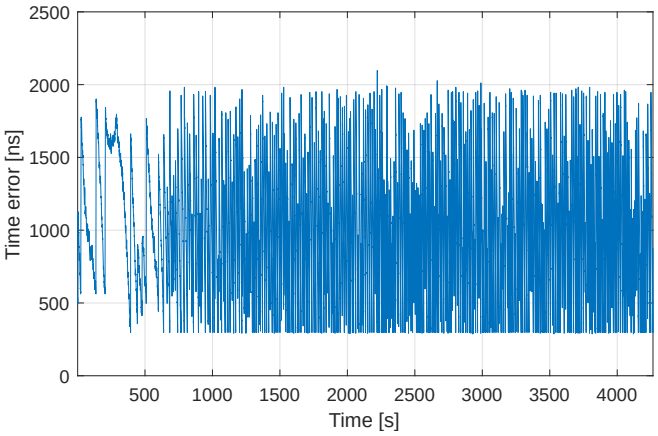


Fig. 5: Long term FS time error measurement

B. Test equipment

For the PTP master clock, I used an Intel 82576 PTP-compliant network interface card controlled by the linuxptp software stack. The less accurate timebase installed on the NIC compared to a grandmaster clock's does not effect sampling synchronization.

For generating the triangle wave test signal I used a HP 3314A function generator.

Tests were done using two nodes featuring the same hardware and software. Collected data have been transferred to a PC and have been plotted using Matlab.

C. Results

The result of the long-term PTP-measurement are shown in fig. 4, results of the long-term FS time error measurement are displayed in fig. 5. At the beginning of the FS measurement plot, a random frequency noise of the crystal oscillator can be observed likely due to some thermal biasing. Error values shown on the fig. 5 only displays FS-to-1PPS error, error values do not have the PTP-error added, there's no significant mathematical correlation between PTP error and FS-to-1PPS error. (PTP errors also can not be determined in every FS synchronization cycle.)

Fig. 6 is a close-up on a single sample point from a triangle wave near zero crossing. Horizontal offset is due to synchronization time error, vertical due to analog circuitry inaccuracies.

IV. Conclusion, Summary

According to the measurements, the system is capable of precise sampling in the range of audio frequencies. Maximum synchronization error between two separate nodes, taking one node's maximal FS time error as 2µs, is approximately 4µs. This level of synchronization is far sufficient for audio frequency range sampling. I made measurements with FS synchronization turned off, the magnitudes of varying phase error, originating from the crystal's inaccuracies were clearly bigger.

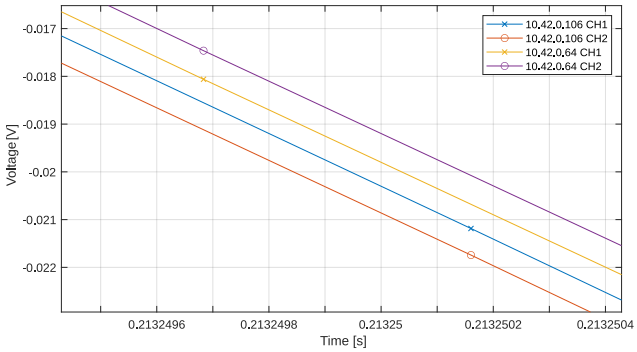


Fig. 6: Close up on a single sample point from a triangle wave measured by two nodes

Although this papers' aim was mainly not to evaluate PTP synchronization stability, the PTP clock also performed well. The time error inferred from the PTP time error is smaller with an order of magnitude than the FS signal synchronization time error.

References

[1] *IEEE 1588-2008 - IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems*, IEEE Std. 1588, 2008.

[2] *IEEE 802.1BA - Audio Video Bridging (AVB) Systems*, IEEE Std. 802.1BA, 2011, https://www.ieee802.org/1/pages/802.1ba.html.

[3] *IEEE 802.1AS-2020 - IEEE Standard for Local and Metropolitan Area Networks–Timing and Synchronization for Time-Sensitive Applications*, IEEE Std. 802.1AS, 2020.

[4] STMicroelectronics, "Stm32h743zi 32-bit arm® cortex®-m7 480mhz mcus, up to 2mb flash, up to 1mb ram, 46 com. and analog interfaces, datasheet - production data," 2021, https://www.st.com/resource/en/datasheet/stm32h743zi.pdf.

[5] ——, "Um1974 user manual stm32 nucleo-144 boards," 2020, https://www.st.com/resource/en/user_manual/um1974-stm32-nucleo144-boards-mb1137-stmicroelectronics.pdf.

[6] TexasInstruments, *TLV320AIC23B Low-Power Stereo CODEC with HP Amplifier*, 2004, https://www.ti.com/lit/ds/symlink/tlv320aic23b.pdf.

[7] A. Wiesner and T. Kovácsházy, "Portable, ptp-based clock synchronization implementation for microcontroller-based systems and its performance evaluation," in *2021 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control, and Communication (ISPCS)*, 2021, pp. 1–6.

[8] PhilipsSemiconductors, "I2s bus specification," 1986, https://www.sparkfun.com/datasheets/BreakoutBoards/I2SBUS.pdf.

Heterogeneous Federated CubeSat System: problems, constraints and capabilities

Carlos Leandro Gomes Batista*, Fatima Mattiello-Francisco*, Andras Pataricza†
*Brazilian National Institute for Space Research,
Space Systems Engineering,
São José dos Campos, Brazil
Email: {carlos.batista, fatima.mattiello}@inpe.br
†Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: pataricza.andras@vik.bme.hu

***Abstract*—Different arguments were being presented in the last decade about CubeSats and their applications. Some of them address wireless communication (5G and 6G technologies) trying to achieve better characteristics as coverage and connectivity.**

Some arrived with terms as IoST (Internet of Space Things), Internet of Satellites (IoSat), DSS (Distributed Space Systems), and FSS (Federated Satellite Systems).

All of them aim to use Small/NanoSatellites as constellations/swarms is to provide specific services, share unused resources, and evolve the concept of satellites-as-a-service (SaS).

This paper aims to emophasize performance attributes of such cyber-physical systems, model their inherent operational constraints and at the very end, evaluate the quality of service in terms of figures of merit for the entering/leaving of new heterogeneous constituent systems, a.k.a satellites, to the constellation. This "whitepaper"-styled work focuses on presenting the definitions of this heterogeneous constellation problem, aims at its main capabilities and constraints, and proposes modeling approaches for this system representation and evaluation.

***Index Terms*—cubesats, constellation, cyber-physical systems, IoSat, IoST, DSS, FSS**

I. Introduction

As the space became more and more accessible, new ways of thinking about the space services have also become more feasible. Issues, like flight formation, constellations, and swarms of satellites were always desirable for the leading space agencies and enterprises. GNSS [1], Sentinel [2], and Iridium [3] are examples of well-established satellites constellation systems, using the coordination between specific purpose developed spacecrafts for global positioning, earth observation and IoT communication, respectively.

The rapid electronics advances, increase of processing capabilities, and low power consumption changed this discussion completely. Since the introduction of the CubeSat standard [4], many works aim the exploration and development of new capabilities for these small satellites [5]. Additionally, the decreased size of these platforms also reduced the launch costs [6], [7]. Now, it is possible to, literally, launch hundreds of less than $5kg$ satellites, each one with entirely different payloads, characteristics, owners, and purposes [8]. All these points lead us towards a new era of re-thinking the idea of satellite constellation systems and their applications.

Funding CAPES process number 88882.444451/2019-01

This work aims to evaluate the primary constraints, problems, and capabilities of such new ideas of using CubeSats in the context of decentralized ownership of heterogeneous satellites cooperating to achieve a common goal, what we can call a Federated CubeSat System.

II. Concepts

Distributed Satellite Systems, DSS, are defined as space systems that allocate functionality through multiple constituent systems to achieve a common goal [9]. These constituent systems are generally different spacecrafts with [10]: (a) same capabilities (constellations and swarms) or; (b) fragmented functionalities, each satellite performs a different activity to achieve the main goal or; (c) decentralized ownership, a federated satellite system (FSS), where different organizations contribute with new satellites and infrastructure.

More specific about the FSS, this kind of system-of-systems establishes the active sharing of unused resources of the multiple constituent systems offered for exploitation in different ways. Shared resources from hosted payloads can service time, processing power, and data rate. Different integration concepts include Internet of Satellites, IoSat, and Internet of Space Things, IoST. They focus on communication and connectivity, relying on the involvement of the spacecrafts on what is called Inter-Satellite Networks, ISN, and Inter-satellite Links, ISL [11].

Key enabling technologies [10], like: (i) dynamic resource allocation and balancing; (ii) power-efficient software-defined radios; (iii) satellite negotiator; (iv) software-defined satellite; (v) virtual space missions, should also be taken into consideration to make these kinds of systems possible.

Once the FSS has implemented the satellite services, its generated payload data can become commodities from the user's point of view, the satellite-as-a-service (SaS). Here, we use the concept of FSS not only for spacecrafts, but we intend

to expand the idea to the whole space system (space, ground, and user segments).

III. Problems, Requirements & Constraints

The major problem in dealing with FSS is its lack of homogeneity [10]. Managing heterogeneity requires some common rules. Starting with a Systems Engineering approach, we must define such a system's primary needs, goals, objectives, and constraints. The goal is to develop measures that will drive each constituent system's impact, with their particular capabilities.

Let us use as an example the Brazilian Environmental Data Collection System, BEDCS, [12] and its exploitation as GOLDS – Global Open coLlecting Data System [13]. The BEDCS and GODLS serve identical overall needs, goals, and objectives, get the data from the Data Collection Platforms (DCP) spread around a specific territory, and send it to the Ground Stations. However, they differ in constraints due to the capabilities of their constituent systems (Figure 1).

For the BEDCS, we have the SCD and CBERS family satellites. These satellites differ in form, design, mission, and primary objective. The CBERS main goal is Earth Observation, and the Data Collection is secondary. Nevertheless, they share the Ground Stations and the hosted payload, which requires simultaneous access/contact with DCP and Ground Station.

The GOLDS, on the other hand, intends to use a new generation of CubeSats from CONASAT and CATARINA constellations beside the SCD and CBERS satellites. Each constellation has its particular objectives, but they will share a common goal for the GOLDS through a new hosted payload, the Environmental Data Collector (EDC), that enables the GOLDS to be global and overcome the simultaneous visibility constraint from BEDCS. Turning the data collection global, we start to deal with new constraints, such as data storage capacity and download rates.

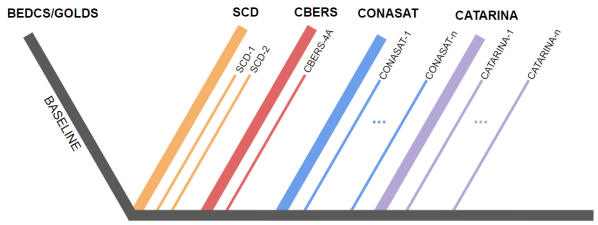


Fig. 1. Baseline and possible composition view of the BEDCS/GOLDS SoS and its constituent systems.

This simple example causes minimal impact on the individual spacecrafts. And it offers services based upon availability of the resources without modifications in the space-to-ground links. We have different satellites cooperating built up by using different technologies, from different organizations [14]. New terrestrial DCPs can enter the network, increasing the demand and the regions of interest, i.e., global. Ways to measure the quality of the service, as FSS, must be considered, even more with the effects of new arriving constituent systems.

A. Setting up the requirements

We can derive some requirements for the constellation from the definition of the goals and objectives of the FSS.

Functional and extra-functional requirements should be derived, but how to do it once we do not control the constituent systems? One approach for that is the approach driven with the unifying concept of operation, ConOps, and its requirements [15]. The federation as a single System of Systems, SoS, has unique characteristics and requirements that the sum of its parts should meet. At the same time, every single participating satellite still keeps its original primary designated mission [16].

Back to the BEDCS/GOLDS example, we can use the ConOps characteristics [15] to define some requirements for the idealized FSS:

- Data Availability
  - The data will be processed onboard for the EDC payload satellites, and on ground for SCD and CBERS satellites.
  - The data must be available as soon as the Mission Center validates the acquired data.
  - All the data is centralized at the Mission Center, located in Brazil.
- Communication Architecture
  - Each satellite must define its particular downlink data rates in compliance with its ground stations.
  - The access link between any satellite and the ground stations must be enough to download all the data from one complete orbital period.
  - The access link between any satellite and one DCP must be enough to upload all the DCP data available.
  - The revisit time (*time between two consecutive overpasses on the same target*) for one DCP must not be more than 1 hour.
- Tasks, Scheduling and Control
  - The use and control of the hosted payload, EDC, must respect the BEDCS/GOLDS decisions and the satellite resources availability.
  - All the federated ground stations must be capable of controlling the hosted payload, EDC.
- Timeline
  - The BEDCS/GOLDS must have the flexibility to receive new satellites with the hosted payload, EDC, to its federation.
  - The BEDCS/GOLDS must have the flexibility to retire satellites from its federation to maintain the federation quality of service.
- Fault Management
  - In case of a fault on an FSS constituent system, the Mission Center must be able to perform a reconfiguration on the FSS resources.
  - The time for reconfiguration must not exceed one operation planning period (*operational activity when all the operational procedures are planned for a*

specific period of time, i.e., overpasses, flight plans, calibration, etc.).

As a System of Systems, an FSS evolves. Its behavior can be defined in terms of its systems independence and interoperability or, to be more specific, retrofitting, which is the capability for systems to interoperate on-demand to meet mission objectives as soon as a new satellite arrives/leaves the federation [17].

B. Constraints

Even knowing the overall objectives of an FSS, some items can remain fuzzy, again, due to the heterogeneity of the constituent systems. How to dynamically integrate unknown resources? Will the system correctly provide the services? Looking at the limitations of our system is, sometimes, more productive. The constraints of the FSS can best formulate the boundaries of its solution space.

The requirements presented earlier can be refined into more detailed requirements and resource constraints. Data availability requirements create constraints on each constituent satellite on data storage and processing. Communication Architecture characteristics derive constraints on bandwidth and data rates to download DCP data. Power consumption and federation engagement time on available resources constrain satellite control and tasks scheduling. The deployment/retirement of new satellites requires a capability of reconfiguration and retrofitting on the FSS. Fault management requirements also influence the FSS configuration.

Extensibility demands the introduction of quality measures into the set of requirements. The respective required minimal and offered values for new constituent satellites decide their integration. , e.g., at least 90% of all DCP coverage, minimum of 10% engagement time for non-dedicated satellites, 2 GB DCP data storage capacity, 2W peak power consumption, one day revisit time, minimum 10 minutes ground station access time per day, at least one dedicated ground station and communication channel [13].

For example, Figure 2 shows the constraint of the BEDCS/GOLDS satellites coverage of DCP.

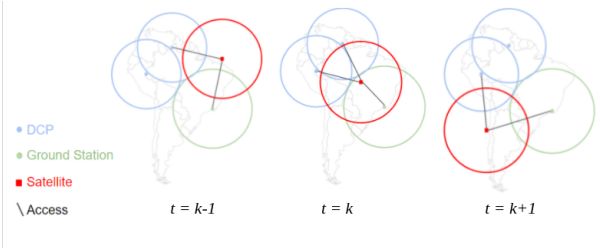


Fig. 2. Coverage Constraint Problem for BEDCS.

Note that, on BEDCS satellites, the access only exists if the satellite can 'view' the DCP and the Ground Station simultaneously, it is a COVERAGE constraint problem.

From now on, we will handle these constraints with the help of the mathematical paradigm called the Constraint Satisfaction Problem (CSP). CSP is an approach that facilitates estimating a single solution, all solutions or the best solution for problems with limitations or conditions, defined into a domain set.

We can define each DCP as a region of interest (ROI) for the satellite and has its field of view (FOV).

$$ROI = [ROI_1, ROI_2, \ldots, ROI_n]$$

The same for the ground station(s):

$$GrSt = [GrSt_1, GrSt_2, \ldots, GrSt_n]$$

The Satellite has its FOV but it changes over time (orbit):

$$Sat = [fov(t)]$$

If we deal with different satellites:

$$Sat = [fov_1(t), fov_2(t), \ldots, fov_n(t)]$$

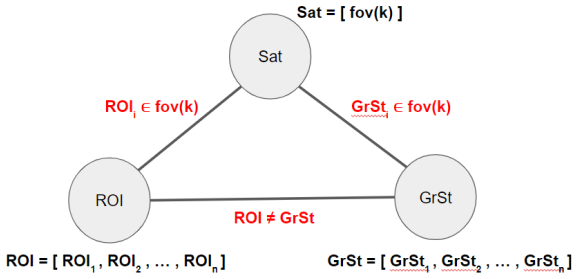


Fig. 3. BEDCS Coverage Constraint Problem Model

Once the constraints are satisfied we have a successful access as can be viewed at the Figure 3.

As the satellite FOV changes over time, we will have an solution for each instant of time.

Given a time interval $t = [0, \ldots, k]$, the sum of these solutions for a specific ROI will give us the COVERAGE characteristic of the satellite for that ROI.

Moreover, if we have a constellation of n satellites, we can derive the constellation COVERAGE for a specific ROI as the sum of each satellite set of solutions.

As we go to the GOLDS concept and the hosted EDC payload, the same problem of COVERAGE transforms itself into a CSP on data storage. The covered DCP networks upload to the satellite an amount of data associated with each ROI:

$$Data = [Data_1, Data_2, \ldots, Data_n]$$

Nevertheless, the satellite has two parameters, FOV and available storage at that specific time, $storage(t)$.

$$Sat = [(fov(t), storage(t))]$$

Again, if we deal with different satellites:

$$Sat = [(fov_1(t), storage_1(t)), ..., (fov_n(t), storage_n(t))]$$

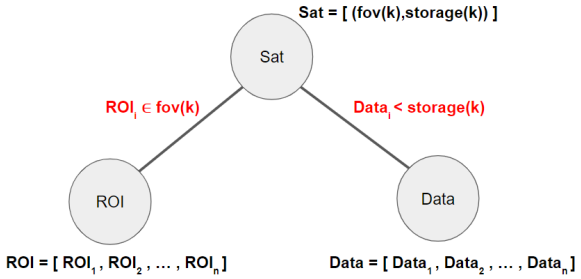


Fig. 4. Data Access as Coverage Constraint Problem Model

Once the constraints are satisfied, we have successful access, as can be viewed in Figure 4.

As the satellite FOV changes over time, we will have a solution for each instant of time.

Given a time interval $t = [0, \ldots, k]$, the sum of these solutions for a specific ROI will give us the COVERAGE characteristic of the satellite for that ROI.

Moreover, if we have a constellation of n satellites, we can derive the constellation COVERAGE for a specific ROI as the sum of each satellite set of solutions.

At the end, the sum of sets of solution for a specific ROI of the BEDCS and EDC satellites give us the COVERAGE of the GOLDS for that specific ROI.

So, evaluating the actual configuration characteristics (i.e. coverage, available satellites, ground stations, storage data, power, revisit time) and the desired configuration to achieve the expected quality of service is a must. This reconfiguring capability configures an FSS emerging behavior by translating into a Constraint Optimization Problem instead of a satisfaction-only problem optimizing service provision with available resources.

IV. Conclusion

Developing a Distributed Satellite System is a challenge task. Validating this idea using CubeSats can be game-changing for the next years. Distributed CubeSat Systems have not yet been demonstrated in large scale, with exception of Planet and Spire over more than 40 proposed constellations and Federated CubeSats concepts have not yet flight.

We propose a possible protocol/process to validate the impacts on a heterogeneous federate CubeSat system of a new satellite or group of satellites deployed in orbit to work in this federation. What is the problems inherent to this kind of system? What to expect from the evolution of this SoS? How much can be modeled once we do not have control over the constituent systems?

Using the BEDCS/GOLDS constellation as an example, we could translate some of the main characteristics of such constellation concepts. We could also start to theorize over this complete satellite system (space, ground, and user segment) as an idea of cooperation and sharing of resources.

Another thing is how to simulate the system. As a dynamic system, the federated CubeSat system and its constituent satellites time dependent, and the fulfillment of their constraints will also change over time. Some tools can be used for that, orbit simulators are well-known but some work is necessary, e.g. NASA General Mission Analysis Tool. Mainly we focus about the representation of the resources available at the system.

We still have significant work to do, not only on the modeling/model side but also on correctly picking the main attributes of this kind of constellation to better formulate the questions we have been asked during this whole paper.

References

[1] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle, *GNSS–global navigation satellite systems: GPS, GLONASS, Galileo, and more*. Springer Science & Business Media, 2007.

[2] E. Attema, P. Snoeij, G. Levrini, M. Davidson, B. Rommen, N. Floury, B. Duesmann, B. Rosich, A. Pietropaolo, V. Mastroddi *et al.*, "Sentinel-1 mission capabilities," *ESA Special Publication*, vol. 677, p. 45, 2010.

[3] K. Maine, C. Devieux, and P. Swan, "Overview of iridium satellite network," in *Proceedings of WESCON'95*. IEEE, 1995, p. 483.

[4] J. Puig-Suari, C. Turner, and R. Twiggs, "CubeSat: the development and launch support infrastructure for eighteen different satellite customers on one launch," *15TH Annual/USU Conference on Small Satellites*, pp. 1–5, 2001. [Online]. Available: http://digitalcommons.usu.edu/smallsat/2001/All2001/59/

[5] National Academies of Sciences, Engineering, and Medicine, *Achieving Science with CubeSats: Thinking Inside the Box*. Washington, DC: The National Academies Press, 2016, no. September. [Online]. Available: http://www.nap.edu/catalog/23503

[6] E. Kulu, "Small Launchers-2021 Industry Survey and Market Analysis," *2021 IAC Proceedings*, vol. 13, no. October, pp. 25–29, 2021. [Online]. Available: www.newspace.im

[7] W. Fox, "Launch costs to low earth orbit, 1980-2100: Future timeline, data & trends, future predictions," accessed: 2021-01-05. [Online]. Available: https://www.futuretimeline.net/data-trends/6.htm

[8] "India launches record 104 satellites at one go," Feb 2017, accessed: 2021-01-05. [Online]. Available: https://www.reuters.com/article/us-space-launch-satellites-india-idUSKBN15U0EI

[9] D. Selva and D. Krejci, "A survey and assessment of the capabilities of Cubesats for Earth observation," *Acta Astronautica*, vol. 74, pp. 50–68, 2012.

[10] C. Cappelletti, S. Battistini, and B. K. Malphrus, *Cubesat Handbook*. Academic Press, 2021.

[11] A. Golkar, "Federated Satellite Systems (FSS): A Vision Towards an Innovation in Space Systems Design," *9th IAA Symposium on Small Satellites for Earth Observation*, 2013.

[12] W. Yamaguti, E. A. Ribeiro, J. C. Becceneri, and S. N. Itami, "Collection and treatment of the environmental data with the brazilian satellite scd1," *Revista Brasileira de Ciencias Mecanicas (ISSN 0100-7386*, vol. 16, pp. 205–211, 1994.

[13] F. J. Vidal *et al.*, "Global open collecting data system (golds): Proposta de requisitos para o desenvolvimento da especificação do sistema," *Revista de Tecnologia da Informação e Comunicação*, vol. 10, no. 1, pp. 1–5, 2021.

[14] A. Golkar and I. Lluch I Cruz, "The Federated Satellite Systems paradigm: Concept and business case evaluation," *Acta Astronautica*, vol. 111, pp. 230–248, 2015. [Online]. Available: http://dx.doi.org/10.1016/j.actaastro.2015.02.009

[15] W. J. Larson and J. R. Wertz, *Space mission analysis and design*. Space Technology Library, 1999, vol. 29, no. 09.

[16] R. Akhtyamov, R. Vingerhoeds, and A. Golkar, "Identifying retrofitting opportunities for federated satellite systems," *Journal of Spacecraft and Rockets*, vol. 56, no. 3, pp. 620–629, 2019.

[17] A. M. Madni and M. Sievers, "System of Systems Integration: Key Considerations and Challenges," *Systems Engineering*, vol. 17, no. 3, pp. 330–347, 2014.

Intrabody Communication Methods – A Short Overview

Matija Roglić, Željka Lučev Vasić
University of Zagreb, Faculty of Electronic Engineering and Computing
Zagreb, Croatia
Email: {matija.roglic, zeljka.lucev.vasic}@fer.hr

***Abstract*—This paper is an introduction to underlying mechanisms and the current state of technologies in the field of intrabody communication (IBC). IBC technologies utilize the human body as a communication channel to achieve communication between different devices that can be positioned inside or on the surface of the body. Current developments in the field of mobile gadgets, smartwatches and medical devices make this field of particular interest due to very low energy expenditure, security, and ability to protect private data. Since there are multiple subfields in the field of IBC technologies, this paper will focus on summarizing the principles of work for galvanic coupling and capacitive coupling and compare the tradeoffs of using one approach over the other.**

Keywords— body channel communication, galvanic coupling, capacitive coupling, biomedical engineering

I. INTRODUCTION

In the recent years there was a steady increase in the number of sold wearables such as smartwatches and wristbands that can track many vital functions such as heart rate, body activity or respiration [1]. These devices usually employ standard wireless data transmissions based on radio frequency (RF) such as Bluetooth, Bluetooth Low Energy and Zigbee, which even though strive for lower power consumption, can still prove inadequate for 24-hour medical usage [2], [3]. The IEEE 802.15.4. standard for low power Zigbee protocol indicates the output power of 0 dBm (1 mW) for transmission at the maximum rate of 250 kb/s which can use up a normal lithium-ion battery in a few hours [4]. The research on military grade equipment that would provide a very low energy consumption for devices that are a part of wireless body area network (WBAN) has stated that the highest battery usage came from RF communication and that significant savings could be made by performing data fusion on the sensor nodes themselves and by reducing the transmission rate [5]. Furthermore, since standard technologies operate at higher frequencies, the range that those devices have is usually much larger than necessary, which wastes energy and can become a security risk for outside intruder attacks such as eavesdropping on the highly private information and even extracting them while impersonating a legitimate WBAN client [4]. In 2007, a former vice-president of the United States Dick Cheney has requested that a wireless control compartment of his defibrillator be removed as he feared that it might be used in a hacking attempt to deliver a fatal electric shock directly to his heart [6].

The concept of using human body as a conductor for electrical current and to transmit information was first explored by Zimmerman [7] who labeled those systems as Personal Area Networks (PAN). PANs would be able to unite multiple in-body and on-body devices without usage of excess cables and I/O redundancies through usage of intrabody communication (IBC). This opens the possibilities of developing medical devices that use IBC to perform communication between each other. For example, an insulin pump could be used in conjunction with a glucose level sensor to measure the levels of glucose in a human body and achieve a continuous transfer of the data to the insulin pump via IBC [8] or to a wearable bracelet which can then display the measured glucose levels to the user [9]. Furthermore, this way of communication can be established between two bodies that are touching, which could enable the transmission of data between two people [10] by simply touching fingers or shaking hands.

There are multiple methods of intrabody communication which have been explored so far that offer a possibility of better characteristics in terms of battery usage, safety, and speed than standard RF communications, such as ultrasound, magnetic resonant coupling, galvanic coupling, and capacitive coupling [9]. Research by Galluccio *et al.* [11] has shown that due to the human body composition which is 65% water, a communication using audio waves at frequencies above 20 kHz (ultrasonic) had significant potential, but issues such as tissue heating and cavitation (bubbles of air within an acoustic field can expand and burst, causing damage to biological tissues) presented that there are still drawbacks that must be further explored. Meanwhile, research by Koshiji *et al.* has proposed usage of loosely fitted coupled coils around parts of human body that could be used to transmit and receive magnetic energy [12]. This method used the property of magnetic fields to freely flow through biological tissue. While these two methods of coupling for intrabody communication have been researched and have their benefits and drawbacks, this paper will focus primarily on two main methods of intrabody communication, which are capacitive and galvanic coupling. Both methods employ a transmitter and a receiver that are battery powered and have a pair of electrodes. They differ in the way electrodes are set up on the human body and the way the signal propagates through the communication channel.

II. INTRABODY COMMUNICATION

A. Capacitive Coupling

Capacitive coupling method utilizes a transmitter and a receiver which consist of two electrodes, a signal electrode which is attached to the human body and a ground electrode which is oriented towards the environment. Both transmitter and a receiver are electrically isolated and battery powered. When a weak electric field is present, human body acts as a signal guide and couples the signal electrostatically [7] while the return signal passes through the environment as shown in Figure 1. These induced electric fields appear between all parts of system that are at different potential. The

DOI 10.3311/MINISY2022-008

transmission signal is generated by modulating the voltage between two signal electrodes and is then received and decoded by the receiver. The induced current through the human body is measured in order of magnitude of picoamperes, and therefore presents no harm to organism [13]. Furthermore, as most of signal is confined to the body while the human body acts as an electric conductor, this minimizes the required transmission power [14]. The received signal level is affected by many factors, such as the signal frequency, position of electrodes, orientation of the transmitter to the receiver, the size of the receiver ground plane, and the surrounding environment. Usual ranges of signal frequencies are between 1 and 100 MHz. Lučev *et al.* [15] have shown that for capacitive coupling in this frequency range the channel gain increases with signal frequency for 20 dB/dec up to around 45 MHz, after which it decreases. But as the frequency increases, the human body starts acting as an antenna and the radiation of signal into the environment is no longer negligible [14].

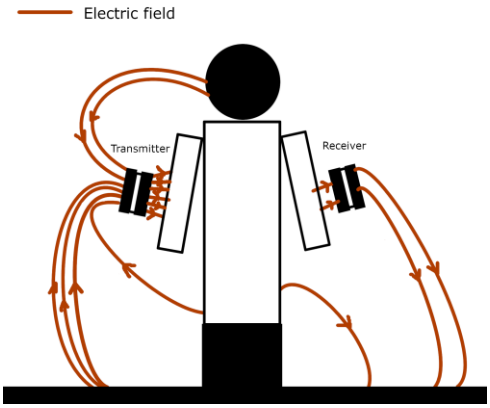


Fig. 1. Capacitive coupling diagram showing the flow of the electrical field from the environment to the electrodes and from electrodes to the human body

Research by Fujii *et al.* has shown that by reducing the size of transmitter by half the received signal voltage was reduced by nearly 50%, compared to only 10% drop when increasing frequency from 10 MHz to 100 MHz [16]. Haga *et al.* have shown that the most sensitive factor for the signal strength maximization was the separation distance between two electrodes, as increasing the distance reduced the capacitance and therefore induced more energy into the body [17]. Furthermore, the size of the ground electrode also had an impact on the transmission gain, with larger electrode increasing it. The gain was also not affected by the size of signal electrode in case it was in contact with the human body. If the signal electrode was not in contact with the human body, then the size of signal electrode also should be maximized to reduce the capacitive loss between the signal electrode and the body. Zhao *et al.* [18] have experimented with design of wristband and found that the best result for wristband was achieved when the separation of two electrodes was maximized, with the signal electrode being in a direct contact with human body and ground electrode being on the top of the wristband. However, for the mobile phones, it was shown that due to huge variety of ways to hold a mobile phone, an optimal design must be found that would prevent the user from short-circuiting the electrodes. Even though the larger sizes of electrode produced better results, the size of wearables and medical instrumentation cannot be too large as it would cause inconvenience or burden for the user. Therefore, a compromise must be found while designing such devices.

When analyzing the effect of body positions on capacitive coupling, Lučev *et al.* have analyzed different positions such as sitting and standing while the transmitter and receiver electrodes were either on the same arm, or on the different arms which increased the distance [15], [19]. Subjects were asked to sit, stand, hold one arm upwards, swing the arm, or hold both arms parallel to the floor. The results showed that for lower frequencies, the body geometry and arm movement had little to no influence on the measured transmission, while for 40 MHz and higher frequencies the change in gain increased to around 20 dB and was influenced by body geometry. Seyedi *et al.* have explored the effect of the limb positions and joints to the transmission gain by placing electrodes on the left forearm and upper left arm [20]. Subject was then asked to stand and perform four different positions with left arm being bent at 45°, 90°, 135°, and 180°, respectively. It was shown that the presence of a joint has attenuated the signal more in frequency range between 60 MHz and 170 MHz. It was also again shown that the position had almost no influence for frequencies lower than 40 MHz in case the distance between transmitter and receiver remained the same, while for higher frequencies the attenuation was proportional to the angle between forearm and upper arm. Sasaki *et al.* have researched the contribution of the ground loop through the floor in IBC by placing the subjects on different types of floors [21]. The subjects wore transmitter or receiver wrists and stood on a carpet-covered metal floor, concrete floor, hardwood floor and wooden chair to be above the floor while touching a receiver on an aluminum stand. The results have showcased that the influence of the ground loop was not significant, as most of signal transmission occurred through the capacitive coupling between the transmitter, the receiver, the human body, and the aluminum stand.

Hou *et al.* have designed an IBC system based on capacitive coupling capable of transmitting image data [22]. By using on-off keying (OOK) modulation with a 20 MHz carrier signal, they achieved a stable image transmission through the body at a rate of 445 kbps. They have also achieved the transmission between two bodies using the handshake as a contact point. OOK modulation was used as it outputs no carrier when transmitting low level and therefore saves energy which makes it suitable for IBC systems which aim to achieve low power consumption. Cho *et al.* have designed a transceiver which can achieve 80 Mb/s data rate with half duplex communication or 40 Mb/s data rate for full duplex communication, with an energy consumption of 79 pJ/b [23]. As described in the paper, this transceiver has been designed to support two modes of operation: the entertainment mode in which the speed and full-duplex is prioritized and healthcare mode with ultra-low-power consumption and high Q-factor. Recently, it was shown that a stable capacitive return path can be accomplished even in implantable devices in case the ground electrode is isolated from the human tissue [24].

B. Galvanic Coupling

Galvanic coupling was first observed by Handa *et al.* in 1997 [25]. Like capacitive coupling, it uses human body as a transmission channel for signal propagation, but the signal return path in galvanic coupling is confined fully within the human body. In galvanic coupling both signal and ground

electrodes must be in contact with the body on transmission and receiving side. Primary current flows between the transmitter electrodes, while a small portion of it propagates

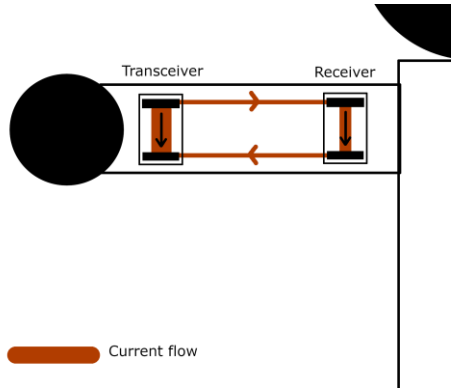


Fig. 2. Galvanic coupling between transmitter and electrode on a human arm

through human body and can be detected as an alternating potential difference between the receiver electrodes [13], as seen in Figure 2. The propagation of the signals through human body is possible due to the relative permittivity and electrical conductivity of the body [26]. Same as capacitive coupling, galvanic coupling also offers low power consumption and low frequency signals. Galvanic coupling is advantageous in that the signal path loss is lower within the human body than it is through the air. This means that the return signal path is not affected by the environment as it is in capacitive coupling. However, due to attachment of low impedance ground electrode in proximity of signal electrode, the signal attenuation is larger. The signal attenuation also increases with the distance [27].

Wegmueller *et al*. have compared different electrodes that can be used for galvanic coupling by generating a 1 mA current signal modulated in the frequency range of 10 kHz to 1 MHz. The tested electrodes were *Swaromed REF 1008*, *Neuroline 715* and *Blue Sensor BR*. For *Blue Sensor* electrodes, different sizes of electrodes were also tested. In conclusion, the electrodes with a lower resistance led to a lower attenuation. If the electrode was smaller, the measured resistance was larger, and the capacitance lower, therefore larger electrodes proved to be better for galvanic coupling. Furthermore, solid-gel electrodes with lower capacitive values have achieved better results than pre-gelled electrodes [28]. Wegmueller *et al*. also kept the transmitter and receiver electronically isolated by using battery powering of the transceiver units and a serial optical connection to connect the units via a universal serial bus (USB) to a standard computer. This way, the units were entirely electrically decoupled from any power lines and adhered to safety limits [29].

Galvanic coupling signals can travel through skin, muscle and fat tissues and its properties are affected by the tissue layer that is used as a medium [9]. Research by Song *et al*. has discovered that distribution of electrical potential was mainly confined in the upper layers of skin and fat, while the muscle had relatively lower potential. When transmitting signal through the extremities, the attenuation increased with distance, while the signal attenuation through thorax was relatively independent of location. The signal transmission distance had less influence on signal attenuation at lower frequency ranges such as 10 kHz – 100 kHz, at 100 kHz – 500 kHz the distance started to influence the attenuation and at 500 kHz – 5 MHz range, the signal attenuation was highly influenced by distance. Same as in capacitive coupling, joints acted as a blockage and added an attenuation of more than 10 dB [29], [30]. This was because joints mostly consist of low-conductivity bone tissue, with less high-conductivity muscle and fat tissue.

Galvanic coupling uses low frequency ranges, usually between 10 kHz and 1 MHz [14]. Since low frequencies cannot support high bandwidth, Asan *et al*. have proposed usage of fat tissue to achieve low loss microwave communication in frequency range of 1.7 GHz to 2.6 GHz [31], [32]. The measurements have shown that loss was around 2 dB per 20 mm in phantom, while in *ex vivo* model the loss was around 4 dB per 20 mm. Compared to transmission of the same frequency signal through the muscle tissue, the loss was two times lower. The optimal thickness of fat layer for signal transmission was 25 mm, however the research did note that variation in body composition has not been considered and the signal quality might be affected in patients with low body mass index as thickness of fat layer decreases.

Usefulness of galvanic coupling in field of practical application was already demonstrated in several cases. Vizziello *et al*. have transmitted electromyography data in both *ex-vivo* and *in-vivo* tissues using galvanic coupling [33]. The achieved SNR for 9 cm distance was 20.45 dB and the performance was almost error free, therefore showcasing the reliability and robustness of this type of communication. This application of galvanic coupling could be used to transmit sensed signals from a healthy muscle to a close one that is unable to receive natural input signals due to a nerve compression. Hachisu *et al*. have designed bracelets that were worn around wrists and could detect whether a contact was made with another person who was also wearing the bracelet [10], [34]. The bracelets contained a three-axis digital acceleration sensor, and the acquired values were then used to determine the type of contact. The bracelet could recognize with an accuracy over 85% if the contact has been conducted with a handshake or only fingertips, and if only with fingertips, with how many. The bracelets could also glow if the contact was made. Noormohammadi *et al*. proposed an ultra-low-power communication approach between an implant and an on-body device [35]. A carrier-less signal was used to reduce power consumption. The total power consumption for this method was 45 µW for the data rate of 64 kb/s. The bit error rate was less than 0.5% for 14 cm distance in homogenous medium and less than 2.5% for 10 cm distance in a multilayer and complex medium without any channel coding techniques. This showed that galvanic coupling could be used for communication with implants that were placed inside the body.

III. Conclusion

In this paper a short overview of two main methods of IBC have been described: capacitive and galvanic coupling. Both methods showcase the possibility of using human body as a communication channel for transmission of data using small amount of energy which is important for medical devices that must operate 24 hours a day. While capacitive coupling can achieve higher transmission rates than galvanic coupling, it is much more influenced by the environment. Galvanic coupling on the other hand despite slower speed achieves safer transmission of private data as there is no

leakage of signals. Moreover, both methods were already used to achieve continuous, low error communication without using error correction algorithms, therefore proving their robustness and reliability as communication methods. More research is needed in these fields as to explore the possibilities of usage of such communication methods in medicine and in other fields.

ACKNOWLEDGMENT

The work of the doctoral student Matija Roglić has been fully supported by the "Young researchers' career development project — training of doctoral students" of the Croatian Science Foundation ESF-DOK-2021-02.

REFERENCES

[1] Statista, "Wearables." 2021. [Online]. Available: https://www.statista.com/study/15607/wearables-statista-dossier/

[2] S. Adibi, "Link technologies and BlackBerry mobile Health (mHealth) solutions: A review," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, no. 4, pp. 586–597, 2012

[3] M. Seyedi, B. Kibret, D. T. H. Lai, and M. Faulkner, "A survey on intrabody communications for body area network applications," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 8, pp. 2067–2079, 2013

[4] K. R. Narayana and S. Saha, "A Certificate Less Encryption and Signature Scheme with Efficient Revocation for Securing Inter-Body Wireless Sensor Network," *International Journal of Science Technology & Engineering*, vol. 2, no. 11, pp. 721–731, 2016, [Online]. Available: www.ijste.org

[5] J. J. Kang, W. Yang, G. Dermody, M. Ghasemian, S. Adibi, and P. Haskell-Dowland, "No Soldiers Left Behind: An IoT-Based Low-Power Military Mobile Health System Design," *IEEE Access*, vol. 8, pp. 201498–201515, 2020

[6] T. Porter, "Dick Cheney Had Heart Device Replaced Over Fears of Homeland Style Terrorist Attack," International Business Times, 19 October 2013. [Online]. Available: https://www.ibtimes.co.uk/dick-cheney-heart-attack-terrorist-homeland-pacemaker-515159. [Accessed 4 January 2022].

[7] T. G. Zimmerman, "Personal Area Networks. Near-field intrabody communication," *IBM SYSTEMS JOURNAL*, vol. 35, no. 3 & 4, pp. 609–617, 1996.

[8] M. Swaminathan, F. S. Cabrera, J. S. Pujol, U. Muncuk, G. Schirner, and K. R. Chowdhury, "Multi-Path Model and Sensitivity Analysis for Galvanic Coupled Intra-Body Communication Through Layered Tissue," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 10, no. 2, pp. 339–351, Apr. 2016

[9] W. J. Tomlinson, S. Banou, C. Yu, M. Stojanovic, and K. R. Chowdhury, "Comprehensive Survey of Galvanic Coupling and Alternative Intra-Body Communication Technologies," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 2, pp. 1145–1164, Apr. 2019

[10] T. Hachisu and K. Suzuki, "Interpersonal Touch Sensing Devices using Inter-Body Area Network," *IEEE Sensors Journal*, pp. 1–1, 2021

[11] L. Galluccio, T. Melodia, S. Palazzo, and G. E. Santagati, "Challenges and implications of using ultrasonic communications in intra-body area networks," in *2012 9th Annual Conference on Wireless On-Demand Network Systems and Services, WONS 2012*, 2012, pp. 182–189.

[12] F. Koshiji, N. Yuyama, and K. Koshiji, "Electromagnetic characteristics of body area communication using electromagnetic resonance coupling," in *2013 IEEE 2nd Global Conference on Consumer Electronics, GCCE 2013*, 2013, pp. 63–64.

[13] Ž. Lučev, I. Krois, and M. Cifrek, "Intrabody Communication in Biotelemetry," *Springer*, vol. 75, pp. 351–368, 2010.

[14] D. Naranjo-Hernández, A. Callejón-Leblic, Ž. L. Vasić, M. Seyedi, and Y. M. Gao, "Past results, present trends, and future challenges in intrabody communication," *Wireless Communications and Mobile Computing*, vol. 2018. Hindawi Limited, 2018.

[15] Ž. Lučev, I. Krois, and M. Cifrek, "A capacitive intrabody communication channel from 100 kHz to 100 MHz," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 12, pp. 3280–3289, 2012

[16] K. Fujii and K. Ito, "Evaluation of the received signal level in relation to the size and carrier frequencies of the wearable device using human body as a transmission channel," in *IEEE Antennas and Propagation Society, AP-S International Symposium (Digest)*, 2004, vol. 1, pp. 105–108.

[17] N. Haga and K. Ito, "Fundamental Characteristics of Electrodes for Intra-Body Communications," in *EuCAP*, 2011, pp. 3623–3626.

[18] K. Zhao, Z. Ying, and S. He, "Intrabody Communications Between Mobile Device and Wearable Device at 26 MHz," *IEEE Antennas and Wireless Propagation Letters*, vol. 16, pp. 1875–1878, 2017

[19] Ž. Lučev, I. Krois, and M. Cifrek, "Effect of Body Positions and Movements in a Capacitive Intrabody Communication Channel from 100 kHz to 100 MHz," in *IEEE International Instrumentation and Measurement Technology Conference*, 2012

[20] M. Seyedi, T. Huei, D. Lai, and M. Faulkner, *Limb Joint Effects on Signal Transmission in Capacitive Coupled Intra-Body Communication Systems*. 2012

[21] K. Sasaki, N. Arai, D. Muramatsu, and F. Koshiji, "Evaluation of Ground Loop Through the Floor in Human Body Communication," *International Journal of Wireless Information Networks*, vol. 24, no. 2, pp. 78–90, Jun. 2017

[22] Y. Hou *et al.*, "Design of Image Transmission System of Intra-Body Communication Based on Capacitive Coupling," 2019.

[23] H. Cho *et al.*, "A 79 pJ/b 80 Mb/s full-duplex transceiver and a 42.5μW 100 kb/s super-regenerative transceiver for body channel communication," *IEEE Journal of Solid-State Circuits*, vol. 51, no. 1, pp. 310–317, Jan. 2016

[24] I. Čuljak, Ž. Lučev Vasić, H. Mihaldinec, and H. Džapo, "Wireless Body Sensor Communication Systems Based on UWB and IBC Technologies: State-of-the-Art and Open Challenges," *sensors*, 2020,

[25] T. Handa, S. Shoji, S. Ike, S. Takeda, and T. Sekiguchi, "A Very Low-Power Consumption Wireless ECG Monitoring System Using Body as a Signal Transmission Medium," in *International Conference on Solid-State Sensors and Actuators*, 1997, pp. 1003–1006.

[26] M. Seyedi, B. Kibret, D. T. H. Lai, and M. Faulkner, "A survey on intrabody communications for body area network applications," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 8, pp. 2067–2079, 2013

[27] J. Jang, J. Bae, and H. J. Yoo, "Understanding Body Channel Communication: A review: From history to the future applications," in *Proceedings of the Custom Integrated Circuits Conference*, Apr. 2019, vol. 2019-April

[28] M. S. Wegmueller, M. Oberle, N. Felber, N. Kuster, and W. Fichtner, "Signal transmission by galvanic coupling through the human body," *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 4, pp. 963–969, Apr. 2010

[29] M. Wegmueller *et al.*, "Measurement system for the characterization of the human body as a communication channel at low frequency," in *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, 2005, vol. 7 VOLS, pp. 3502–3505.

[30] Y. Song, K. Zhang, Q. Hao, L. Hu, J. Wang, and F. Shang, "A finite-element simulation of galvanic coupling intra-body communication based on the whole human body," *Sensors*, vol. 12, no. 10, pp. 13567–13582, Oct. 2012

[31] N. B. Asan *et al.*, "Intra-body microwave communication through adipose tissue," *Healthcare Technology Letters*, vol. 4, no. 4, pp. 115–121, 2017

[32] N. B. Asan *et al.*, "Data packet transmission through fat tissue for wireless intra body networks," in *IEEE Journal of Electromagnetics, RF and Microwaves in Medicine and Biology*, Dec. 2017, vol. 1, no. 2, pp. 43–51.

[33] A. Vizziello, P. Savazzi, and G. Magenes, "Electromyography Data Transmission via Galvanic Coupling Intra-body Communication Link," Sep. 2021

[34] T. Hachisu, B. Bourreau, and K. Suzuki, "EnhancedTouchX: Smart bracelets for augmenting interpersonal touch interactions," May 2019.

[35] R. Noormohammadi, A. Khaleghi, and I. Balasingham, "Galvanic Impulse Wireless Communication for Biomedical Implants," *IEEE Access*, vol. 9, pp. 38602–38610, 2021.

Investigating the combined application of Mendelian Randomization and constraint-based causal discovery methods

Mihály Vetró, Márton Bendegúz Bankó, Gábor Hullám
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: {vetro, bankom}@edu.bme.hu, gabor.hullam@mit.bme.hu

***Abstract*—Mendelian randomization (MR) is often used in medical studies and biostatistics, to reveal direct causation effects between exposures and diseases, typically the effect of some exposure (like chemicals, habits and other factors) to a known disease or disorder. However, this procedure has some strict prerequisites, which often do not comply with the known variables, or the exact causal structure of the variables is not known in advance. In this study, we investigate the use of constraint-based causal discovery algorithms (PC, FCI and RFCI) to produce a sufficient causal structure from the known observations, to aid us in finding variable triplets, upon which MR can be performed. In addition, we show that the validity of MR cannot always be determined based on its results alone. Finally, we investigate the application of the MR principle to determine the direction of causality between variable-pairs, which is a problem most constraint-based causal discovery methods struggle with.**

***Index Terms*—Mendelian Randomization, Bayesian networks, constraint-based causal discovery, causal effect strength, biostatistics**

I. Introduction

In this study, we investigate three constraint-based causal discovery methods, and Mendelian Randomization (MR), which is a well-known method for causal effect estimation in biostatistics and medical studies, and then we examine two different approaches to combine them. A similar study including genetic anchors has been performed by Howey et al. [1], in which they investigated some simple causal structures regarding MR. Here, we take a more general approach with regards to the size and number of the investigated causal graphs, and we also examine the usability of the MR principle to help determine the direction of causality in uncertain cases.

II. Mendelian randomization

MR can be classified as a local causal discovery method applied in the field of genetic studies. However, while observational data based general causal discovery methods learn the causal structure from data with no prior assumptions regarding the structure, MR methods are based on a predefined causal structure relying on a set of assumptions which need to be fulfilled [2]. The MR model (causal structure) is a causal chain formed by a triplet of variables ($G \rightarrow E \rightarrow D$) which consists of the following elements:

- Genetic variant (G): the gene whose effect is being studied.
- Exposure factor (E): an event, occurrence or influencing factor to which susceptibility is influenced by a genetic variant, and that factor has an effect on the disease.
- Disease (D): the diagnosis itself, which may be influenced by the previous factors.
- Confounding factor (U): additional variable that is not part of the chain but may affect the exposure and disease variables.

MR methods use the effect size (e.g. log odds ratio in case of categorical variables) between the gene - exposure (β_{GE}) and the gene - disease (β_{GD}) variables to infer the magnitude of the effect between the exposure and the disease (β_{ED}) as: $\beta_{ED} = \frac{\beta_{GD}}{\beta_{GE}}$, which can be treated as a Wald ratio and its significance can be determined accordingly [3]. A significant ratio can be considered as an indication that the causal effect between the exposure and the disease is significant and that the causal relationship $E \rightarrow D$ exist.

The assumed MR structure (displayed in Fig. 1) encodes the following assumptions:

- A1: The association between the genetic variant and the exposure factor should be strong. In its absence, the strength of the MR is reduced and bias may occur.
- A2: The genetic variant is independent of the confounding factor. Otherwise, the confounding factor would affect both the disease and the genetic variant, which may imply that the gene-disease effect detected by the method is only indicative of the difference due to the confounding factor.
- A3: The disease is conditionally independent of the genetic variant given the exposure factor. It follows that the gene does not directly influence the presence of the disease, instead it only has a mediated effect.

The main question of MR methods is how to ensure this specific structure shown in Fig.1. In general, it requires

This research was supported by the ÚNKP-21-5-BME-362 New National Excellence Program of the Ministry for Innovation and Technology from the source of the National Research, Development and Innovation Fund, and the János Bolyai Research Scholarship.

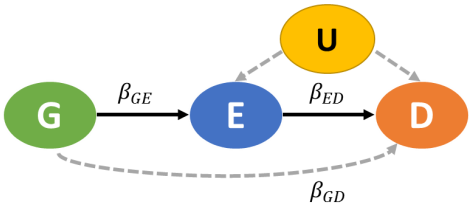


Fig. 1. The assumed MR model.

considerable background knowledge to exclude variables from the dataset that do not satisfy the validity assumptions. The main weakness of the MR method is its simplistic, rigid model. Although it can be efficient when the investigated relationships are simple, i.e. there are no confounding or interacting factors, in more complex cases however, the assumptions of the MR model are unrealistic. This either leads to the inability to use MR methods in several real-word scenarios or to an inappropriate application of MR disregarding some of the validity assumptions. To address this issue, the basic model has been extended in several ways to make the method more robust: MR Egger [4], MR-link [5]. However, the main feature of causal structure learning algorithms, i.e. the structure is learned from the data, is still missing from MR methods.

III. Causal structure learning methods

We selected three widely-known constraint-based methods to investigate their integrated application with a MR method: the Peter-Clark (or PC) algorithm, Fast Causal Inference (FCI) and *Really* Fast Causal Inference (RFCI).

A. The Peter-Clark algorithm

The Peter-Clark (PC) algorithm is a local method [7], relying on examining variable pairs to determine if they are (conditionally) dependent, and variable triplets - or more precisely, chain structures containing exactly 3 variables - to determine the direction of causality between the dependent variables. The former step leads to a skeleton, i.e. an undirected graph of dependency relationships which can be facilitated by applying conditional independence tests on variables. The latter step requires the detection of triplet-based uniquely identifiable dependency structures, called V-structures [6] ($X \rightarrow Z \leftarrow Y$), whose edges can be unambigously directed. The second step leaves those edges undirected that are not part of a V-structure, which may include a significant number of edges. Additional steps using various heuristics may be applied to orient these undirected edges.

B. Fast Causal Inference

While the PC algorithm is built to reconstruct the full causal graph of the variables, the Fast Causal Inference (FCI) [8] and its more efficient version, the *Really* Fast Causal Inference (RFCI) algorithm [9] both aim to reconstruct the equivalence class of the original causal structure, represented by its essential graph, which is a partially directed acyclic graph (PDAG).

IV. Approach No. 1: Augmenting MR with causal structure learning

In our first approach, we investigated the usefulness of the causal discovery methods described in section III. to determine if MR is applicable for a given Gene-Exposure-Disease variable triplet. In our methods, if the three variables form a directed chain in the same order ($G \rightarrow E \rightarrow D$), then it is considered as a valid candidate for MR, and considered invalid otherwise.

First, we examined some simple causal structures, shown in Fig. 2. For these models, the PC, FCI and RFCI algorithms were all capable to reliably reconstruct the original causal graph from at least 1000 samples. This is the expected result, because almost all of the edges are part of at least one V-structure, apart from the edge ($3 \rightarrow 4$) in Model 1 and the edges ($3 \rightarrow 4$) and ($3 \rightarrow 5$) in Model 3. In our tests, all the variables were binary, which represents a discrete variable case of MR. Note that it is also possible to apply MR for continuous disease score and exposure variables. The conditional probabilities were sampled randomly from a uniform distribution. From the simpler graphs, the invalid triplets (where at least one of the $G \rightarrow E$ and $E \rightarrow D$ edges were missing) produced similar Wald-ratios (which are estimations[1] for β_{ED} given by $\beta_{ED} = \frac{\beta_{GD}}{\beta_{GE}}$) to the valid triplets. This suggests that there are certain cases, in which the applicability of MR cannot be determined by the estimation on β_{ED} alone.

To investigate this more generally, we made 50 randomly generated causal graphs, using a simple stochastic algorithm, which iteratively generated random parent-sets for every node (selected from the previously visited nodes), thus creating a guaranteed DAG. Out of the 50 models 25 models had 5 Gene, 3 Exposure and 2 Disease variables (10 in total), while the other 25 had 15 Gene, 10 Exposure and 5 Disease variables (30 in total). The 25 smaller graphs had 6.6 valid and 23.4 invalid paths on average (30 possible paths in total), while the 25 larger models had 20 valid and 730 invalid paths on average (750 possible paths in total). This level of sparsity is roughly representative of the true causal structures of real-world datasets containing Genetic, Exposure and Disease variables. To examine the results of MR, we are only concerned with the estimated strength of causal effect between the Exposure and Disease variables. This value is higher, if β_{ED} is far from 0 in any direction (positive or negative), therefore it is appropriate to use the absolute value of β as a measure for the strength of causal effect. The resulting $|\beta|$ values for the randomly generated partitioned graphs are presented in Table I. From these results, it is evident that the expected value ($\mathbb{E}$) and standard error (σ) of $|\beta_{ED}|$ is significantly larger for invalid triplets, compared to the valid ones.

[1]To estimate the β (effect size) between supposed cause-effect variable pairs, we used logistic regression with 35 steps and the Newton-Raphson optimizer, yielding logarithmic odds-ratios ($log(OR)$).

This can be explained by a simple phenomenon: if we take four variables: $\{X, Y, Z, W\}$, where $(X \to Y \to Z)$ form a valid triplet, so Y is strongly dependent on X, and Z is strongly dependent on Y, but W has no causal connection to any of the other three variables. Therefore, the value of $|\beta_{XZ}|$ will be high (because X affects Z through Y), but the value of β_{XW} will be close to 0, because they are independent. As a result, if we (wrongly) perform MR on the $\{X, W, Z\}$ invalid triplet, then we will get a high value for $|\beta_{WZ}| = |\beta_{XZ}/\beta_{XW}|$. Because of this, we can get a significantly higher $|\beta|$ value for invalid triplets even compared to the valid ones, therefore the use of causal discovery methods are well justified to rule out the invalid cases.

In terms of predictive performance, all three methods were able to find on average 50% of the valid triplets in our 50 partitioned models at 15.000 samples, with a precision of 98%, which means, that 98% of the predicted triplets were correct.

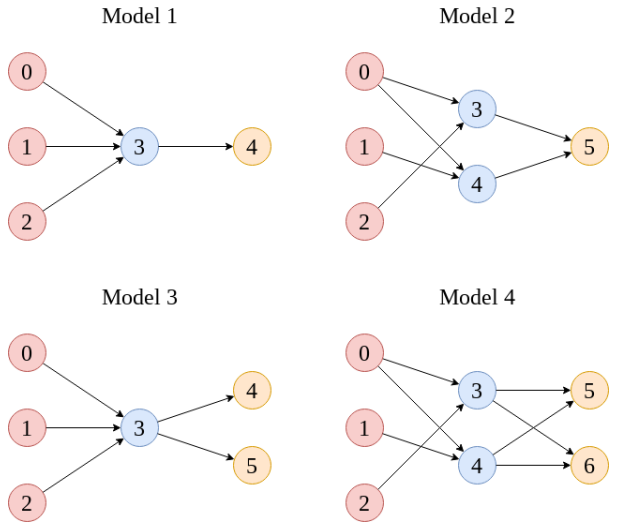


Fig. 2. Models used to demonstrate typical MR β-values. The gene, exposure and disease variables are marked accordingly with red, blue and yellow colors.

V. Approach No. 2: Orienting undirected edges using the MR principle

As we have discussed before, the FCI and RFCI algorithms produce a partially directed graph, where the undirected edges are assumed to be undirected – by the algorithm – in the original essential graph, which belongs to the equivalence class of the original causal structure. In other words, the algorithm assumes that these edges cannot be directed given the known data. Although, within real-world datasets, the number of known samples are finite, and often susceptible to noise. Because of this, we assume that the predicted essential graph will not be perfect, therefore in some cases, the edges that are left undirected by the FCI and RFCI algorithms can be directed by investigating the possible candidate triplets for MR, which the edge in question is a part of. To examine this theory, we propose a method, that consists of the following steps:

1) Acquire a partially directed acyclic graph $\mathcal{G}$ from the known samples using an arbitrary constraint-based structure learning algorithm.
2) For every undirected $(X - Y)$ edge in $\mathcal{G}$, search for all the possible genetic variables, which are not already invalidated by the known directed edges. This includes all the neighbors of X and Y, which are either their parents or they are connected to either of them by an undirected edge. Let's mark the set of these candidate variables by $\boldsymbol{C}_X$ and $\boldsymbol{C}_Y$ for the neighbors of X (not including Y) and the neighbors of Y (not including X) respectively.
3) For every undirected $(X - Y)$ edge in G, find the best candidates for genetic variables G_X and G_Y (in terms of both directions), which are given by:

$$G_X = \arg\max_{G_X \in \boldsymbol{C}_X} \left| \frac{\beta_{G_X Y}}{\beta_{G_X X}} \right| \quad G_Y = \arg\max_{G_Y \in \boldsymbol{C}_Y} \left| \frac{\beta_{G_Y X}}{\beta_{G_Y Y}} \right| \quad (1)$$

4) Use G_X to calculate β_{XY} and G_Y to calculate β_{YX}. If $\beta_{XY} > \beta_{YX}$ then orient the edge as $X \to Y$, otherwise orient the edge as $X \leftarrow Y$

This method basically finds the best possible MR triplet for both directions, and orients the edge at the direction determined by the triplet with the highest $|\beta_{ED}|$ score. While the PC method does not give a direct estimation to the essential graph of the original causal structure, it does not provide a direction for most of the edges in the skeleton which are not part of an V-structure. Therefore, we will also examine the applicability of the above described method for the edges that are left undirected by the PC algorithm.

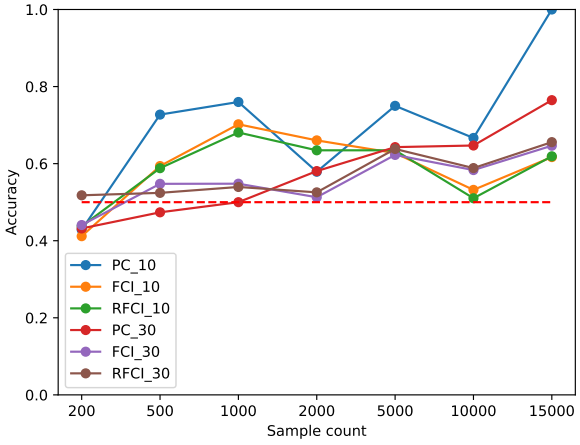


Fig. 3. Edge orientation accuracy (on undirected edges) by the MR β metric, across multiple causal discovery algorithms on partitioned causal graphs with 10 and 30 nodes, with $log(OR)$ beta values. Note, that in case of the unoriented edges of the PC algorithm for the 10-node partitioned graphs, a large number of edges could not be oriented by MR, because they had at least one 0 value in their contingency table, therefore its results are not significant for these graphs. This also explains the outlying accuracy numbers of PC_10 compared to FCI_10 and RFCI_10.

TABLE I
ABSOLUTE β-SCORES ON VALID AND INVALID EXPOSURE-DISEASE TRIPLETS

	Variable count			Log Odds-Ratio			
				Valid		Invalid	
Sample count	**G**	**E**	**D**	$\mathbb{E}(\lvert\beta\rvert)$	$\sigma(\lvert\beta\rvert)$	$\mathbb{E}(\lvert\beta\rvert)$	$\sigma(\lvert\beta\rvert)$
500	5	3	2	0.84	4.21	5.05	36.39
1000	5	3	2	0.69	2.43	5.16	18.57
5000	5	3	2	0.31	0.51	14.29	143.59
10000	5	3	2	0.32	0.98	8.89	37.49
15000	5	3	2	0.32	0.74	10.81	47.06
500	15	10	5	0.66	1.57	4.25	23.81
1000	15	10	5	0.91	4.76	7.08	181.81
5000	15	10	5	0.70	5.56	10.05	286.69
10000	15	10	5	0.45	4.20	6.96	60.45
15000	15	10	5	0.23	0.34	8.64	124.34

In terms of results, the orientation accuracy of our method on the edges left undirected by the PC, FCI and RFCI algorithms on the 50 randomly generated partitioned graphs can be seen in Fig. 3. These results indicate, that our method can predict the orientation of the undirected edges at above-chance levels, from at least 1000 samples. Note, that the partitioned nature of the original causal graph is not assumed by the causal discovery algorithms, and neither by our method, because of which most of the edges oriented by our method are not actually between supposed Exposure-Disease variables. If that were the case, the accuracy would be significantly higher. However, this is an assumption which we cannot make without prior knowledge about the variables, which is not always available. In the partitioned graphs with 10 nodes and 15.000 samples, on average 34% of the edges predicted by FCI and RFCI were undirected, while this ratio rose to 41% in the partitioned graphs with 30 nodes with both algorithms. The PC algorithm left 44% of the edges undirected in the partitioned graphs with 10 nodes, and this ratio fell to 38% for the partitioned graphs with 30 nodes. If we orient these edges randomly (with an even distribution), then in the partitioned graphs with 10 nodes, the FCI and RFCI algorithms oriented 74% of *all* predicted edges correctly, while this accuracy raises to 77% on average with both algorithms, if we used our method to orient the undirected edges. In case of the unoriented edges of the PC algorithm for the 10-node partitioned graphs, a large number of edges could not be oriented by MR, because they had at least one 0 value in their contingency table, therefore the results were not significant. This also explains the outlying metrics of PC_10 in Fig. 3. In case of the partitioned graphs with 30 nodes and also 15.000 samples, MR improved the orientation accuracy of the FCI and RFCI methods from 67% (with random edge orientation) to 72% (with MR edge orientation). However, on these 30-node graphs, the edge orientation accuracy of PC only marginally improved to 74% with MR, compared to the 73% that the algorithm would produce with random undirected edge orientation.

Finally, for the sake of completeness, we also examined the edge orientation performance of this method on completely random directed acyclic graphs (which are therefore not partitioned). Unsurprisingly, it did not produce the same above-chance accuracy values seen on partitioned graphs, further supporting our belief that it only works on the second edge of valid Gene-Exposure-Disease triplets.

VI. CONCLUSION

In this study, we showed that the validity of MR cannot necessarily be determined based on its result, therefore it is advisable to use causal discovery methods for this purpose. We also showed, that MR (for a certain class of directed acyclic causal structures) can improve the edge-orientation capability of the PC, FCI and RFCI methods in terms of the edges that are left unoriented by the original algorithm. As further research, we plan to investigate the integration of MR into other types of causal discovery algorithms, like score-based methods.

REFERENCES

[1] Howey, R., Shin, S. Y., Relton, C., Davey Smith, G. and Cordell, H. J. (2020). Bayesian network analysis incorporating genetic anchors complements conventional MR approaches for exploratory analysis of causal relationships in complex data. PLoS genetics, 16(3), e1008198.

[2] Burgess, S., Foley, C.N., Allara, E., Staley, J.R. and Howson, J.M., (2020). A robust and efficient method for Mendelian randomization with hundreds of genetic variants. Nature communications, 11(1), pp.1-11.

[3] Teumer A. (2018). Common Methods for Performing MR. Front. Cardiovasc. Med, Volume 5, 10.3389/fcvm.2018.00051.

[4] Bowden, J., Davey Smith, G., Haycock, P.C. and Burgess, S., 2016. Consistent estimation in MR with some invalid instruments using a weighted median estimator. Genetic epidemiology, 40(4), pp.304-314.

[5] van der Graaf, A., Claringbould, A., Rimbert, A., Westra, H.J., Li, Y., Wijmenga, C. and Sanna, S., 2020. Mendelian randomization while jointly modeling cis genetics identifies causal relationships between gene expression and lipids. Nature communications, 11(1), pp.1-12.

[6] Pearl, J. (2000). Causality: Models, Reasoning and Inference, C. U. P.

[7] Spirtes, P. and Glymour, C. (1991). An Algorithm for Fast Recovery of Sparse Causal Graphs. Social Science Computer Review - SOC SCI COMPUT REV. 9. 62-72. 10.1177/089443939100900106.

[8] Spirtes, P. (2001). An Anytime Algorithm for Causal Inference. Proc. of the 8th Int. Workshop on AI and Statistics, PMLR R3:278-285, 2001.

[9] Colombo, D., Maathuis, H. M., Kalisch, M. and Richardson, S. T. (2012). Learning high-dimensional directed acyclic graphs with latent and selection variables. The Annals of Statistics, 10.1214/11-aos940.

Physical Activity Recognition Based on Machine Learning

Krunoslav Jurčić, Ratko Magjarević
Department of Electronic Systems and Information Processing
University of Zagreb, Faculty of Electrical Engineering and Computing
Zagreb, Croatia
krunoslav.jurcic@fer.hr, ratko.magjarevic@fer.hr

***Abstract*—The following paper presents a comparison study of various machine learning techniques in recognition of activities of daily living (ADL), with special attention being given to movements during human falling and the distinction among various types of falls. The motivation for the development of physical activity recognition algorithm includes keeping track of users' activities in real-time, and possible diagnostics of unwanted and unexpected movements and/or events. The activities recorded and processed in this study include various types of daily activities, such as walking, running, etc., while fall activities include falling forward, falling backward, falling left and right (front fall, back fall and side fall). The algorithm was trained on two publicly available datasets containing signals from an accelerometer, a magnetometer and a gyroscope.**

***Index Terms*—biomedical engineering, physical activity, machine learning, signal processing, accelerometer, magnetometer, gyroscope, fall detection, fall distinction, activities of daily living**

I. Introduction

In recent times Human Activity Recognition (HAR) has become a research area that attracted a great deal of interest, especially in biomedicine and biomedical engineering. One of the reasons for that is the availability of data: today user data are available through many devices, such as smartphones and their apps, portable sensors, wristbands, smartwatches, etc. for real-time analysis. These sorts of analysis can provide useful information in a wide range of applications such as irregularity detection and correction in various motions, injury prevention, monitoring of progress in rehabilitation, monitoring of athletes' progress, elderly care, etc.

In this paper various machine learning methods were applied and compared on two publicly available datasets, with ultimate goal of providing a trustworthy algorithm for recognition of various activities of daily living (ADL), such as e.g. walking, running, jumping among others. Aside from ADLs, an additional analysis was conducted, with the emphasis being on detection and distinction of irregular motions, in this case that being human falls (FALL). One of the aforementioned applications of the results obtained in this particular study is fall recognition among the elderly population. With the COVID-19 pandemic outbreak, people, and especially the elderly, were forced to spend more time at their homes. According to United Nations World Population Ageing 2020 Highlights, historical data show that the living arrangements of elder persons have changed slowly over time, shifting from co-residence towards independent living [1]. In this case, real-time fall prediction could provide the elderly living alone with the necessary help in case of falling.

II. Methodology

A. Datasets

1) PIV dataset: The first dataset used in this study is a publicly available dataset by Pires, Garcia, Zdravevski and Lameski, called "Polytechnic Institute of Viseu (PIV) dataset" [2]. It contains recordings of five activities of daily living: walking, running, standing, walking upstairs and walking downstairs. The dataset consist of data acquired from accelerometer, magnetometer and gyroscope sensors, recorded by a mobile device in a waistband placed on subject's waist. A total of 25 individuals (15 men, 10 women) aged between 16 and 60 took part in this data acquisition. 10 of those subjects declared themselves as physically active, whilst the rest described their lifestyle as mainly sedentary. The signals acquired from accelerometer and gyroscope sensors were sampled with a frequency f_s of 100 Hz, while the signals acquired from magnetometer have a sampling frequency of 50 Hz. The dataset contains 34.037 records, with the duration of each recorded signal being approximately five seconds. This dataset is publicly available for download and processing.

2) UniZg activ2 dataset: The second dataset used in this study for classification of activity recognition methods was recorded by the Faculty of Electrical Engineering and Computing, University of Zagreb by Razum, Šeketa, Vugrin and Lacković [3]. The activities were recorded using Shimmer3 inertial measurement unit (IMU), using its built-in triaxial wide range accelerometer with a range of +/- 8g, triaxial magnetometer and triaxial gyroscope sensors with a sampling frequency f_s of 204.8 Hz [4]. Figure 1 shows a visual representation of a subject wearing the device. 19 subjects aged 15 to 44 wore the aforementioned device attached to their waist with a Velcro belt and performed nine activities of daily living ("standing", "sitting down", "walking", "standing up", "walking downstairs", "walking upstairs", "lying down", "running" and "jumping") and three activities of simulated

DOI 10.3311/MINISY2022-010

falls on 2 cm thick tatami mat (”falling forward”, ”falling backward” and ”side falling”). That brings to a total of 1.607 signals describing 12 classes for prediction. The waveforms of accelerometer signals describing each activity are presented in figure 2. By looking at the waveforms one can notice potential classification difficulties, since some activities show similar waveforms, e.g ”sitting”, ”standing up” and ”lying down”.

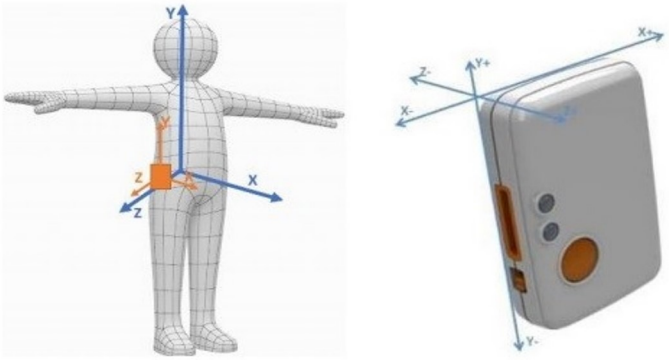


Fig. 1. Computer-simulated visual representation of a subject(left) wearing Shimmer3 inertial measurement unit (IMU) (right)

B. Feature extraction and preprocessing

From the measured raw acceleration data, a sum vector magnitude (SVM) was calculated to combine the data from X, Y and Z axis into one signal using the following equation:

$$SVM(n) = \sqrt{a_x^2(n) + a_y^2(n) + a_z^2(n)} \quad (1)$$

After calculating the sum vector magnitude of every recording, various features were calculated for both raw signal data and SVM signal data. List of features includes the following metrics:

- mean value
- minimum value of signal
- maximum value of signal
- median
- variance
- standard deviation
- signal range
- kurtosis
- skewness
- signal energy

After calculating the values of features for each recorded signal and its SVM signal, an important step towards building a model with better results of classification presents preprocessing of the calculated data. In this study preprocessing consisted of two stages: scaling and dimensionality reduction. For scaling in this study we used *StandardScaler* function implemented in Python *sklearn.preprocessing* library. As far as dimensionality reduction goes, we used ANOVA (Analysis of variance) test which is commonly used when dealing with multiple classes classification problems. Using the ANOVA test we reduced the number of relevant features from initial 100 (10 features x 3 sensors x 3 axes + 10 SVM signal features) to 32, therefore also reducing the risk of overfitting.

TABLE I
CLASSIFICATION ACCURACY FOR *PIV* DATASET

	SVM	RF	DT	GNB	AB	KNN
Acc	**95.12**	**96.45**	**92.75**	**80.73**	**70.84**	**93.86**
Mag	81.91	84.20	75.12	55.36	54.98	78.56
Gyr	87.72	91.57	86.60	60.39	61.86	86.34
Acc + mag	87.96	89.79	83.11	62.78	58.33	86.69
Acc + gyr	91.44	93.40	87.45	67.61	56.18	90.00
Mag + gyr	84.59	87.73	79.64	54.63	53.59	82.55
Acc + mag + gyr	87.90	89.64	82.71	56.88	59.95	86.47

TABLE II
CLASSIFICATION ACCURACY FOR *UniZg activ2* DATASET

	SVM	RF	DT	GNB	AB	KNN
Acc	77.42	**87.10**	**77.06**	32.97	57.71	66.67
Mag	58.06	70.97	50.00	**69.35**	58.06	46.77
Gyr	67.74	83.87	75.81	58.06	**70.97**	**79.03**
Acc + mag	74.71	76.76	71.18	32.65	53.53	63.82
Acc + gyr	**81.72**	81.47	67.06	32.65	56.76	64.41
Mag + gyr	69.35	71.77	69.35	53.23	56.45	56.45
Acc + mag + gyr	71.64	78.11	66.67	32.34	57.21	63.93

III. RESULTS AND DISCUSSION

For the ADL and fall recognition classifier model training we used several machine learning algorithms: Support Vector Machine (SVM) with both linear kernel and radial basis function (RBF) kernel, Random Forest Classifier (RF), Decision Tree Classifier (DT), Gaussian Naïve Bayes (GNB), Adaptive Boosting Classifier (AB) and *K*-Nearest Neighbours (KNN) algorithm with *k*=5. Some of these methods were selected based on results achieved by Ivascu, Cincar, Dinis and Negru implementing the same machine learning algorithms, although using different features [5]. The datasets were divided into training data (70% for the *PIV* dataset, and 75% for *UniZg activ2* dataset) and testing data (30% for the *PIV* dataset, and 25% for *UniZg activ2* dataset). In order to maximize the accuracy of the results, hyperparameter tuning was performed for every machine learning method used in this research. With the previous knowledge of the type of each activity, evaluation of the models accuracy was conducted using the *F*-score metric, more precisely F_β-score. F_β-score is calculated using the following equation:

$$F_\beta = \frac{(1+\beta)^2 * P * R}{\beta^2 * P + R}, \quad (2)$$

where *P* represents precision, and *R* represents recall. The aforementioned, more general, version of the F-score uses a positive real factor β, where β is chosen such that recall is considered β times as important as precision. In this case β=1 is used. Tables I and II show F_1 score results for the two datasets used in this study.

This classification problem was trained on various types of data, including training of data from a solitary sensor

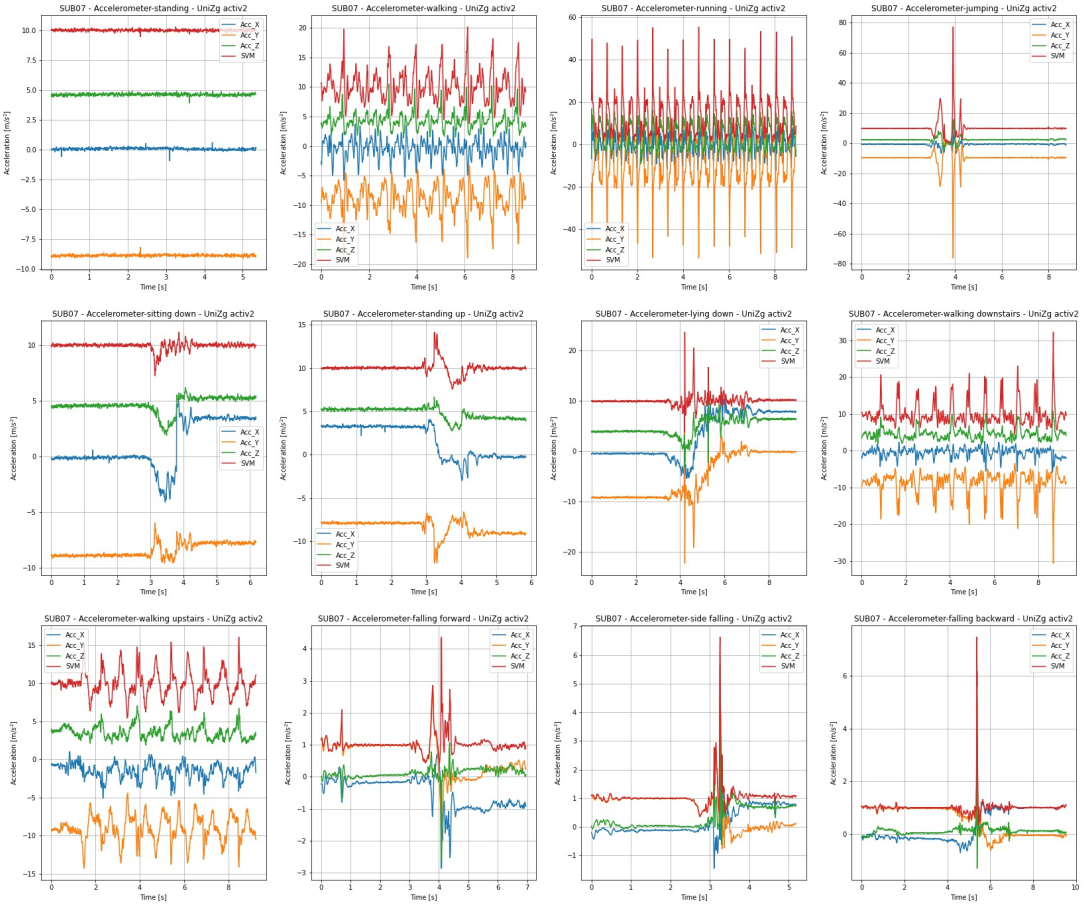


Fig. 2. Visualization of physical activities - *UniZg activ2* dataset, in sequence from left to right: standing, walking, running, jumping, sitting down, standing up, lying down, walking downstairs, walking upstairs, falling forward, side falling and falling backward

("Acc", "Mag, "Gyr"), a combination of two sensors' data ("Acc + mag", "Acc + gyr", "Mag + gyr"), and finally including signals acquired from all three sensors ("Acc + mag + gyr"). A comparison of various types of data shows the variation in each machine learning technique's performance when dealing with them. As it is shown in Tables I and II, training of acceleration data usually perform with the best results, while magnetometer data in most cases perform the poorest. The classification of recordings of *PIV* dataset using each machine learning algorithm used in this study performs best when dealing with acceleration data. *UniZg activ2* dataset classification however varies. Accelerometer and gyroscope data, and the combination of the two, mostly performed best, depending on the machine learning algorithm. Regarding the activities prediction, and with the information presented in Tables I and II, Random Forest Classifier (RF), the best-performing algorithm was chosen for the visualization of the accuracy of each activity detection. The confusion matrices demonstrating the results of classification are shown in Figures 3 and 4. All of the results shown in these figures are performed using only accelerometer data.

Comparing the accuracy of ADL prediction using the first and the second dataset it shows that models perform better when dealing with the dataset that contains more signal recordings, which helps the model with generalization and describes a lower number of activities, therefore deals with fewer classes C for classification.

The greatest number of false positive results among *PIV*

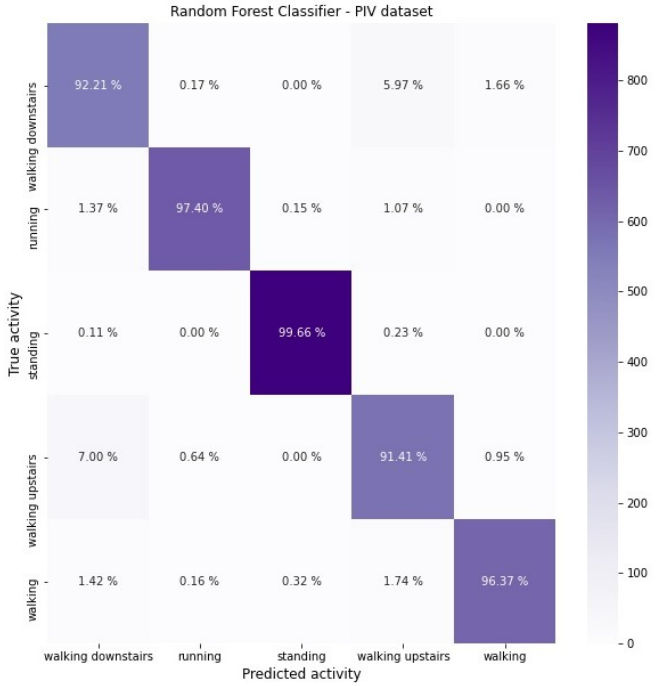


Fig. 3. Random Forest Classifier results - *PIV* dataset

Fig. 4. Random Forest Classifier results - *UniZg activ2* dataset

dataset activities were performed among ”walking downstairs” and ”walking upstairs”, therefore these activities were classified with the poorest accuracy, even though the accuracy still exceeded 90 percent. Regarding the *UniZg activ2* dataset, the algorithms performed better classification when dealing with activities of longer duration (”walking” and ”running”) and activities of greater intensity (”jumping” and ”falling”). Activities of shorther duration and lower intensity and signal energy (”lying down”, ”standing up” and ”sitting down”) were classified with a poorer accuracy.

Another area of research we dedicated a great deal of attention was distinction of different types of falls [6]. Confusion matrices presented in Figures 5 and 6 show the accuracy of machine learning algorithms when distinguishing among three types of falls using two best-performing machine learning algorithms: Support Vector Machine (SVM) and Random Forest Classifier (RF). The aforementioned confusion matrices show that fall distinction problems were well performed by both algorithms. It is important to stress the fact that, even though they show promising results, for achieving higher reliability, validation on datasets larger than currently available *UniZg activ2* dataset should be performed. The authors are aware of the limits of the experiments performed. Although fall detection system are mostly aimed for the use in elderly care, participants involved in this study were mostly younger subjects. Falls were simulated in safe laboratory settings, and although the participants were instructed to fall relaxed, it is possible that those falls may differ from real-life falls. However, real-life daily activity and fall labeling can prove to be a difficult task due to the fact that it is very time consuming, and also because of the issue of user discipline.

IV. Conclusion and Future Work

The main goals of this research were finding the best possible solutions for classification of daily activities and unexpected events, in this case falls. Classifications were conducted using two different datasets using various machine learning techniques. Aside from ADL recognition, fall detection and fall distinction were areas monitored with great interest. Although there is a limited amount of fall data recordings, the predictions made by Random Forest Classifier and Support Vector Machine algorithms have shown their potential for further research in this area. For future work, aside from expanding *UniZg activ2* dataset with more simulated fall recordings, it would be in the best interest of this research to include real life records of accidental falls of elderly, since fall detection algorithms' main application would be primarily elderly care. As far as increasing prediction accuracy of some activities that showed poor results (”sitting down”, ”standing up”), the authors have proposed sensor fusion of the sensors used for recording the *UniZg activ2* dataset with a barometric altimeter. With an increased amount of data, new more complex methods of artificial intelligence, such as Convolutional Neural Networks (CNN) could be explored as a more efficient solution to this problem.

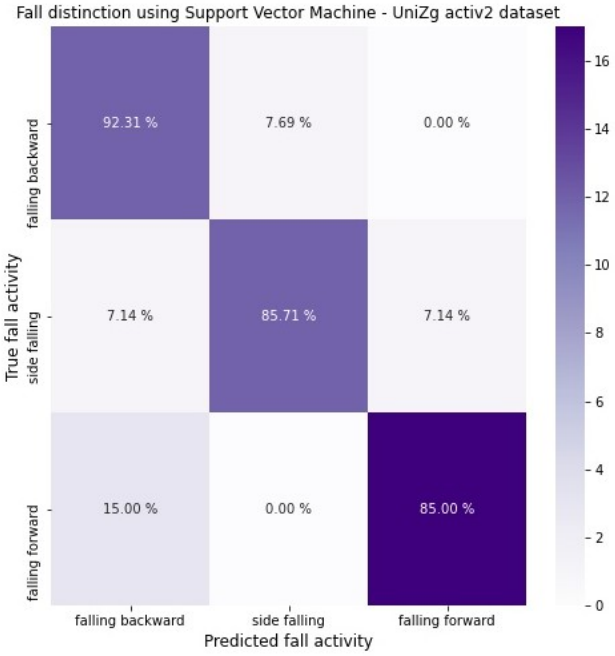


Fig. 5. Fall distinction using Support Vector Machine - *UniZg activ2* dataset

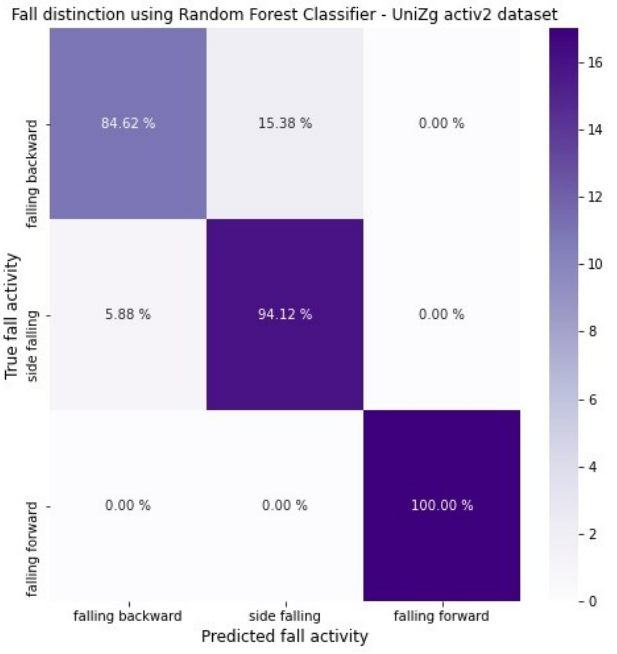


Fig. 6. Fall distinction using Random Forest Classifier - *UniZg activ2* dataset

V. Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] [Online] "United Nations - World Population Ageing 2020 Highlights" URL: https://www.un.org/development/desa/pd/sites/www.un.org.development.desa.pd/files/undesa_pd-2020_world_population_ageing_highlights.pdf

[2] Pires IM, Garcia NM, Zdravevski E, Lameski P. Activities of daily living with motion: A dataset with accelerometer, magnetometer and gyroscope data from mobile devices. Data Brief. 2020 Dec 8;33:106628. doi: 10.1016/j.dib.2020.106628. PMID: 33344738; PMCID: PMC7735969.

[3] D. Razum, G. Seketa, J. Vugrin and I. Lackovic, "Optimal threshold selection for threshold-based fall detection algorithms with multiple features," 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2018, pp. 1513-1516, doi: 10.23919/MIPRO.2018.8400272.

[4] [Online] "Shimmer3 GSR+ Unit" URL: https://shimmersensing.com/product/shimmer3-gsr-unit/

[5] T. Ivascu, K. Cincar, A. Dinis and V. Negru, "Activities of daily living and falls recognition and classification from the wearable sensors data," 2017 E-Health and Bioengineering Conference (EHB), 2017, pp. 627-630, doi: 10.1109/EHB.2017.7995502.

[6] Šeketa, G.; Pavlaković, L.; Džaja, D.; Lacković, I.; Magjarević, R. Event-Centered Data Segmentation in Accelerometer-Based Fall Detection Algorithms. Sensors 2021, 21, 4335. https://doi.org/10.3390/s21134335

Pole Optimization of IIR Filters using Backpropagation

Kristóf Horváth, Balázs Bank
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: {hkristof, bank}@mit.bme.hu

***Abstract*—Audio signal processing is a field where specialized techniques are used to account for the characteristics of hearing. In filter design the resulting transfer function need to follow the specification on an approximately logarithmic frequency scale, which can be done via methods such as frequency warping or fixed-pole parallel filters. Although these IIR filter design techniques are proven in practice, they do not produce optimal pole sets for the given specification. In this paper we present the first experiments of using a gradient-based pole optimization framework implemented in TensorFlow by realizing the IIR filter as a recurrent neural network (RNN). The method can improve the pole set of a filter compared to the initial pole set, resulting in a smaller approximation error. The proposed method is demonstrated using four example filter specifications.**

***Index Terms*—audio filter design, RNN, IIR filter**

I. Introduction

In audio filtering, infinite impulse response (IIR) filters are commonly used [1], where logarithmic frequency resolution is highly desired, to approximate the characteristics of hearing. In order to achieve this, several structures were developed including warped filters [2] and second-order fixed-pole parallel filters [3].

Warped IIR filters are derived from a direct-form IIR structure by substituting allpass sections into the delay elements [2]. The resulting structure has an additional parameter λ, called warping coefficient. In the design process, the specification is first transformed according to λ and then the filter coefficients are set using traditional methods such as Prony's method or the Steiglitz-McBride algorithm. Figure 1 shows the frequency mapping of the warping transformation. In essence the warping coefficient controls the frequency resolution of the filter by making specific parts of the specification more dominant in the warped frequency domain.

In fixed-pole parallel second-order filters the frequency resolution is controlled by the setting the poles appropriately. After that, the filter response becomes linear in the numerator parameters and thus they can be estimated using the least squares (LS) method [4]. In their simplest form, fixed-pole parallel filters have predetermined pole sets which control their frequency resolution – for example poles uniformly distributed on logarithmic frequency scale will result in logarithmic frequency resolution. In addition, several more sophisticated pole positioning strategies have been developed which offer better modeling accuracy at the expense of a somewhat more complicated filter design procedure [5], [6].

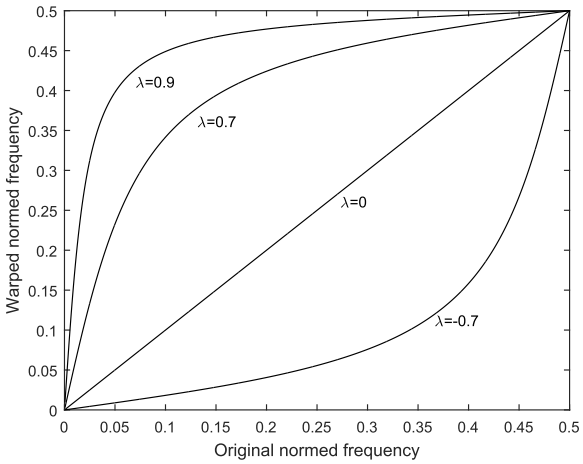


Fig. 1. Frequency transformation of warping.

In this paper we investigate whether the performance of fixed-pole parallel filters can be improved by optimizing their pole set using the backpropagation algorithm.

II. IIR filters as recurrent neural networks

A recurrent neural network (RNNs) is a class of artificial neural networks, which is often used in natural language processing. Contrary to the commonly used feedforward neural network topologies, RNNs have internal memory (state) and assume that the input has one temporal dimension, which can be arbitrarily long. The input is therefore processed along the time dimension. There are many commonly used nonlinear RNN structures such as long short-term memory (LSTM), gated recurrent unit (GRU), Elman network, etc.

In essence all linear, time invariant IIR systems can be considered as a special case of RNNs. To illustrate this, let's consider the Elman network [7], a simple RNN structure for language processing:

$$\mathbf{h}[n] = \sigma_h(W_h\mathbf{h}[n-1] + U_h\mathbf{x}[n] + \mathbf{b}_h), \quad (1)$$

$$\mathbf{y}[n] = \sigma_y(W_y\mathbf{h}[n] + U_y\mathbf{x}[n] + \mathbf{b}_y), \quad (2)$$

DOI 10.3311/MINISY2022-011

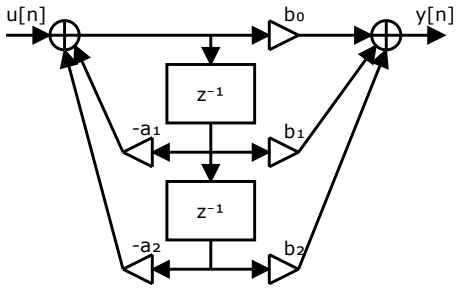


Fig. 2. IIR direct-form 2 second-order section.

where the vectors **x, y, h** are the layer inputs, outputs and hidden states, respectively. The matrices W_h, W_y, U_h, U_y and vectors $\mathbf{b}_h, \mathbf{b}_y$ are the trainable weights of the network, n is the temporal dimension, while σ_h, σ_y refer to activation functions, which are usually nonlinear functions. By setting the bias vectors to $\mathbf{b}_y = \mathbf{b_h} = 0$, and removing the nonlinearities, the resulting equations are equivalent to the state-space representation of IIR filters:

$$\mathbf{h}[n] = W_h\mathbf{h}[n-1] + U_h\mathbf{x}[n], \quad (3)$$

$$\mathbf{y}[n] = W_y\mathbf{h}[n] + U_y\mathbf{x}[n]. \quad (4)$$

By representing audio filters as RNNs, the tools and frameworks for training neural networks become accessible for filter design [8]. When using mean square error (MSE) as cost function, the training process will converge to the least squares solution.

The state space representation of IIR filters preserve the filter structure, therefore it's important to specify the format of the matrices and vectors in Equations 3-4. By restricting which matrix elements can be trained the linear dependencies between variables can be eliminated.

The state space representation of a second-order IIR direct-form 2 section, as seen in Figure 2, is the following:

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}[n+1] = \begin{pmatrix} -a_1 & -a_2 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}[n] + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u[n], \quad (5)$$

$$y[n] = \begin{pmatrix} b_1 - b_0a_1 & b_2 - b_0a_2 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}[n] + b_0u[n]. \quad (6)$$

Note that contrary to the notation used in neural networks, in filter structures the (hidden) state is denoted by $\mathbf{x}$, while the input is denoted by u. Thus the state-space IIR direct form representation in Equations (5)-(6) corresponds to the regular transfer function:

$$H(z) = \frac{b_0 + b_1z^{-1} + b_2z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}}. \quad (7)$$

RNNs are trained using backpropagation through time (BPTT), which is a gradient-based technique. A training step consists of two parts: forward propagation and backpropagation. In the former the network outputs and the cost function (also called loss function) are calculated by unrolling the mathematical operations used by the network for each time point. In the second step the derivatives of the loss function are calculated for all network weights. The network weights are updated using the analytically computed gradients such that the loss would decrease in each iteration.

III. Proposed method for pole optimization

Training the filter using backpropagation is equivalent to a gradient descent method with analytically computed gradients. Contrary to Prony and Steiglitz-McBride methods [9] though, the optimization problem in this case is nonconvex, assuming both the poles and zeros are tuned. This can cause problems such as the tendency to stuck in a local minimum as well as being prone to instability.

Iterative optimization methods, such as backpropagation, are sensitive to the initial network weights. Incorrect setting of the initial values can lead to slow convergence or getting stuck in a local optimum. Therefore we suggest that the initial network weights should be designed using the Steiglitz-McBride algorithm.

Because in Prony and Steiglitz-McBride methods designing the poles of the filter provide the biggest challenge, we suggest that only the poles of the filter should be trained using backpropagation and after a few epochs the zeros should be updated via least squares (LS) in one step. This training cycle should be repeated a few times.

Considering the previous points, the algorithm to design audio filters is the following:

1) **Warp the specification.** As a first step the filter specification is transformed to the warped frequency domain, where the filter design is performed [2].
2) **Design IIR filter using Steiglitz-McBride algorithm.** The Steiglitz-McBride method provides the initial values that are close to the optimum.
3) **Convert filter to parallel structure.** In order to use the previously designed poles and zeros, the coefficients of the direct-form representation are converted to parallel second-order representation using partial fraction expansion [10].
4) **Optimize poles with backpropagation.** The previously computed coefficients are set as the network weights of a filter represented as an RNN. In this structure only the coefficients related to the poles are trained with MSE as cost function.
5) **Design zeros for optimized poles with LS.** After the poles are optimized, the zeros are adjusted using a one-step least squares method based on Moore-Penrose pseudoinverse [4].
6) **Repeat steps 4-5.** In each iteration the remaining MSE is lowered with diminishing returns.
7) **Dewarp the second-order denominators.** In order to implement the filter, the coefficients designed in the warped frequency domain need to be transformed back. Because the dewarping would insert an additional zero to the second-order sections and thus would increase the computational demand, only the denominator is dewarped. For direct-form second-order sections the trans-

formation is performed using the following equations [11]:

$$a_1' = \frac{(1+\lambda^2)a_1 - 2\lambda a_0 - 2\lambda a_2}{1 - \lambda a_1 + \lambda^2 a_2}, \quad (8)$$

$$a_2' = \frac{a_2 - \lambda a_1 + \lambda^2 a_0}{1 - \lambda a_1 + \lambda^2 a_2}, \quad (9)$$

where the dewarped coefficients are denoted by prime.

8) **Design zeros with optimized poles.** Using the optimized pole set, the zeros of the parallel filter are designed with the usual least squares method [4], similarly to step 5.

IV. Ensuring stability

Recurrent neural networks often suffer from two major issues during training: the vanishing and the exploding gradients problem. The former is the result of unrolling nonlinear activation functions in time and as such it it is not relevant for IIR filters represented as RNNs. The exploding gradients problem, however, can be encountered when the poles of the filter are moved outside the unit circle, resulting in an unbounded growth at the output, which leads to high error values. One way to circumvent this issue is to use small learning rate and carefully initialize the coefficients [8].

In order to ensure that the poles would not become unstable during training, we have added a regularizer such that if a pole is moved outside of the unit circle in a training step, the regularizer puts it back to the circle while keeping its frequency intact.

Deriving the formulas for a conditional regularizer is hard for an arbitrarily high degree IIR filter, but for a second-order IIR direct-form section it is easy. Here we show the equations used by our implementation. If

$$4a_2 > a_1^2 \quad (10)$$

then the section has a conjugate complex pole pair. In this case the pole radius is computed as

$$R_c = \sqrt{a_2}. \quad (11)$$

If $R_c > 1$ then the pole is outside the unit circle and must be corrected to avoid instability. Optionally this condition can be tightened to avoid instability after coefficient quantization in the implementation. Thus the condition for correction is $R_c > 1-\epsilon$ where ϵ is a small number that limits the maximum amplification of the pole. The correction is done using these formulas:

$$a_1 := \frac{a_1}{R_c + \epsilon}, \quad (12)$$

$$a_2 := \frac{a_2}{(R_c + \epsilon)^2}. \quad (13)$$

If the condition in Equation (10) is false then the section has real poles. In this case the two poles and their radii are the following:

$$p_{1,2} = \frac{1}{2}\left(a_1 \pm \sqrt{a_1^2 - 4a_2}\right), \quad (14)$$

$$R_{1,2} = |p_{1,2}|. \quad (15)$$

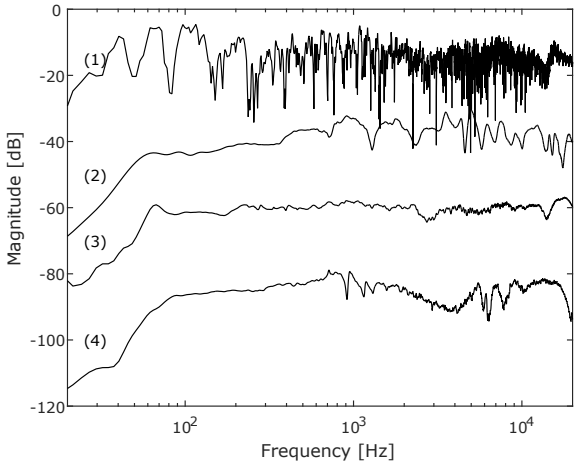


Fig. 3. Magnitude plots of example transfer functions. (1) is a room response, (2)-(4) are loudspeaker responses. The plots have been shifted in order to fit the figure.

System	Prony	Steiglitz-McBride	Proposed method	Gain
1.	$1.49 \cdot 10^{-5}$	$1.15 \cdot 10^{-5}$	$1.05 \cdot 10^{-5}$	9%
2.	$1.19 \cdot 10^{-6}$	$6.82 \cdot 10^{-7}$	$6.00 \cdot 10^{-7}$	12%
3.	$1.97 \cdot 10^{-6}$	$1.47 \cdot 10^{-6}$	$1.24 \cdot 10^{-6}$	16%
4.	$5.48 \cdot 10^{-7}$	$2.19 \cdot 10^{-7}$	$1.78 \cdot 10^{-7}$	19%

TABLE I
Mean square error (MSE) losses of different design methods on the example specifications. The values are calculated in the warped domain. Additionally, the reduction of the MSE error compared to the Seiglitz-McBride method is also shown.

If either of the pole radii are larger than 1, the pole regularizer moves them back to the unit circle. Suppose $R_1 > 1 - \epsilon$, the correction formulas are the following:

$$a_1 := a_1 - p_1 + \frac{p_1}{R_1 + \epsilon}, \quad (16)$$

$$a_2 := \frac{a_2}{R_1 + \epsilon}. \quad (17)$$

The formulas are similar for the case of $R_2 > 1 - \epsilon$.

Note that the zeros of the filter have no direct effect on the stability. The only way a zero can contribute to instability is when it covers a pole that is outside of the unit circle – in this case internal overflow can happen. Restricting the pole movements will eliminate this problem.

V. Experiments

To demonstrate the proposed method, we have designed second-order parallel filters with 4 different specifications, shown in Figure 3: one room and three loudspeaker impulse responses. In this paper we refer to these transfer functions by their numbers.

In our experiments, we have designed parallel filters with 10 second-order sections, altogether 20 poles and zeros. The warping coefficient was $\lambda = 0.9$, the learning rate during back-propagation was 10^{-4}, each pole optimization had 15 epochs and 800 steps per epoch. For the learning rate scheduler

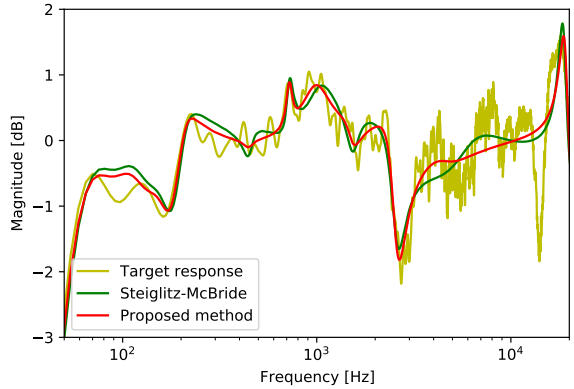


Fig. 4. Magnitude plots of system (3) and the filter responses designed by Steiglitz-McBride method and the proposed method.

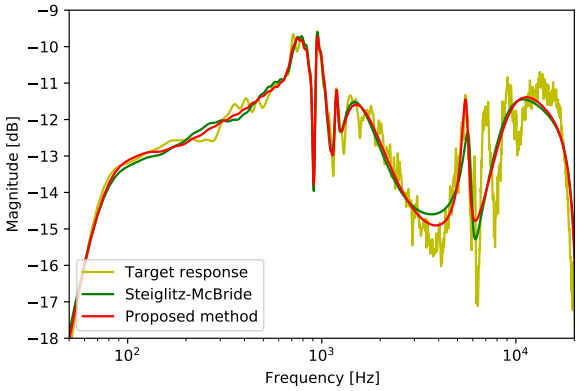


Fig. 5. Magnitude plots of system (4) and the filter responses designed by Steiglitz-McBride method and the proposed method.

we have used Adam [12] with default moments. The whole design process had 5 optimization cycles (steps 4-5 in the algorithm). The scripts were implemented using Python 3.6 and Tensorflow 2.1.0.

The achieved mean square error (MSE) values of the design process, which has been the targets of the optimization process in the warped domain, can be found in Table I. For reference we have added the results of traditional methods such as Prony and Steiglitz-McBride. Note that because the impulse responses are decaying over time and the mean is computed for $N = 1000$ samples, their mean squared value is small, leading to very small error values for all cases. However, this does not mean that this error is negligible, or would be comparable to an error coming from coefficient quantization, for example.

Accordingly, it is not the actual value of the MSE that describes the improvement due to optimization, but the relative reduction of the MSE compared to previous methods. It can be seen that the proposed method can produce coefficients that fit the example specifications with 9-19% lower remaining mean square error compared to traditional methods.

The magnitude plots of the specification and the designed filters can be found in Figures 4-5. It can be seen that the filter designed by Steiglitz-McBride method is improved by the proposed method in the full frequency range.

It should be mentioned that we have found that backpropagation is particularly sensitive to the learning rate. By setting too large learning rates the design process does not converge and therefore does not result in a usable filter. However, when setting the learning rates to too small values the poles barely get shifted and thus the method practically keeps the initial values. Finding a correct learning rate is a process of trial and error.

VI. Conclusion and Future Work

In this paper we have proposed a method for improving the poles of parallel filters using backpropagation. The results show that the method can produce filters that have lower mean square errors compared to the ones based on the original pole set designed by the Steiglitz-McBride method.

Future work includes using the proposed method for equalization, not just for modelling. Since the gradient calculation and the convergence rate is dependent on the structure of the second-order section, different implementation structures should be evaluated as well.

The use of backpropagation opens up the possibility for different cost functions, which can lead to audio filter design methods where the transformation (e.g. warping) is embedded in the loss function.

References

[1] V. Välimäki and J. D. Reiss, "All about audio equalization: Solutions and frontiers," *Applied Sciences*, vol. 6, no. 5, 2016, art. no. 129, doi: https://doi.org/10.3390/app6050129.

[2] A. Härmä, M. Karjalainen, L. Savioja, V. Välimäki, U. K. Laine, and J. Huopaniemi, "Frequency-warped signal processing for audio applications," *J. Audio Eng. Soc.*, vol. 48, no. 11, pp. 1011–1031, Nov. 2000.

[3] B. Bank, "Audio equalization with fixed-pole parallel filters: An efficient alternative to complex smoothing," *J. Audio Eng. Soc.*, vol. 61, no. 1/2, pp. 39–49, Jan. 2013.

[4] ——, "Perceptually motivated audio equalization using fixed-pole parallel second-order filters," *IEEE Signal Process. Lett.*, vol. 15, pp. 477–480, 2008.

[5] ——, "Loudspeaker and room equalization using parallel filters: Comparison of pole positioning strategies," in *Proc. 51st AES Conf. on Loudspeakers and Headphones*, Helsinki, Finland, Aug. 2013.

[6] E. Maestre, G. P. Scavone, and J. O. Smith, "Design of recursive digital filters in parallel form by linearly constrained pole optimization," *IEEE Signal Process. Lett.*, vol. 23, no. 11, pp. 1547–1550, Nov. 2016, doi: https://doi.org/10.1109/LSP.2016.2605626.

[7] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990.

[8] B. Kuznetsov, J. D. Parker, and F. Esqueda, "Differentiable IIR filters for machine learning applications," in *Proc. Int. Conf. Digital Audio Effects (eDAFx-20)*, 2020, pp. 297–303.

[9] K. Steiglitz and L. E. McBride, "A technique for the indentification of linear systems," *IEEE Trans. Autom. Control*, vol. AC-10, pp. 461–464, Oct. 1965.

[10] A. V. Oppenheim, R. W. Schafer, and J. R. Bruck, *Discrete-Time Signal Processing*. Englewood Cliffs, New Jersey, USA: Prentice-Hall, 1975.

[11] K. Horváth, "Rounding effects in audio filters," Master's thesis, Budapest University of Technology and Economics, Budapest, Hungary, Dec. 2015, in Hungarian.

[12] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

Semantically enabled design for edge cyber-physical systems

Nándor Lengyel
Department of Measurement and Information Systems
Budapest University of Technology and Economics
Budapest, Hungary
lengyelnandor@edu.bme.hu

Imre Kocsis
Department of Measurement and Information Systems
Budapest University of Technology and Economics
Budapest, Hungary
kocsis.imre@vik.bme.hu

***Abstract*—Sensor and computational diversity and redundancy enable radically different implementations of the same functionality in the field and edge domains of cyber-physical systems (CPSs). This diversity and redundancy will also become cornerstones of reconfiguration-based resilience in CPSs, but to truly exploit them, the various provided and required services must be matched semantically. We present the prototype of an integrated, semantically enabled design toolset for data flow-centric CPS systems. A data stream graph DSL based on the W3C SOSA/SSN ontologies serves for high-level specification; semantically enabled function allocation in the Stardog knowledge graph platform creates high-level deployment models, exported in the TOSCA format. Based on the data flow semantics, implementation artifacts for the TOSCA model are automatically generated. These constitute Kubernetes containers for stream processing, and OMG DDS or Hyperledger Fabric for data stream graph nodes.**

***Index Terms*—cyber-physical systems, edge computing, deployment toolchain, semantic techniques, TOSCA, Kubernetes**

I. Introduction

Penetration and diversity of network-connected sensors – mobile as well as fixed ones – grow rapidly in our environment. Therefore many contemporary Cyber-Physical Systems (CPSs) can now be composed dynamically. These compositions can be done by adapting sensor ensembles to such changes in the environment as faults, user equipment mobility and changing requirements. Field-to-edge communication with Quality of Service (QoS) guarantees and edge-deployed cloud services ensure that reconfigurable computational resources are also present ”near to the field”.

At the same time, current CPS development and operation practices are ill-suited to achieve even moderate levels of adaptation agility. For instance, a human designer can easily see that e.g. street cameras or a statically deployed radar may be used or combined to determine whether an intersection or railway crossing is free. Such human-made replacement can be relevant in the absence or failure of dedicated sensors.

At the same time, the computer-aided design of CPSs adaptable in this sense has to cope with different data formats and observational metadata (which is sometimes not even explicit). True automated design faces even more serious challenges. It have to reason about the *meaning* of the data delivered by various sensors and the computations over their readings towards a specific CPS goal.

Semantic representation of field device capabilities, requirements and solution design promises to be a fundamental component in achieving computer-aided design support and design automation of edge CPS systems. Such semantic solutions can dynamically compose the sensor ensembles, supporting computational capabilities and cloud services that they rely on [1].

II. Design approach

Our paper explores the feasibility and practicality of such semantically enabled design for *data stream processing* at the edge. To this end, we describe a semantically enabled end-to-end edge-CPS design methodology and toolchain prototype. Figure 1 gives an overview of the overall design approach.

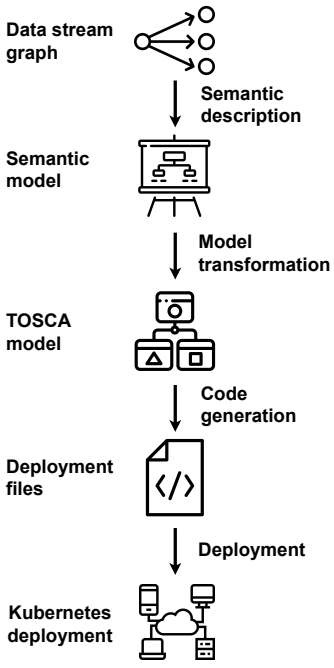


Fig. 1. Design approach overview

Conceptually, the design approach relies on expressing the logical structure of CPS applications as a directed graph of *data stream nodes*, where the edges represent stream element

DOI 10.3311/MINISY2022-012

transformations. The stream elements carried by individual streams are described semantically. Semantic edge annotation with some basic statistical transformations are also supported already. Using data stream graphs, it is possible to look for alternative solutions with graph theory solutions even in the initial phase.

The rationale for emphasizing data streams instead of computation is twofold. On the one hand, the deployment-independent structure of data-intensive CPS applications tends to be defined by the integrating middleware. These middlewares manages the streaming of data from the field through the edge to the cloud. Such as message queuing and publish-subscribe solutions, small-scale (in many cases, embedded and in-memory) databases and increasingly, distributed ledgers. From a model-driven system engineering point of view, a data stream graph expresses this ”logical skeleton” of realized applications far better than data flows. On the other hand, the semantic inference for sensor ensembles and their replacements will also deal with matching the *data available* and the *data required*. Besides that also important,whether we possess the accompanying computational capability, but it is a ”second-step” concern.

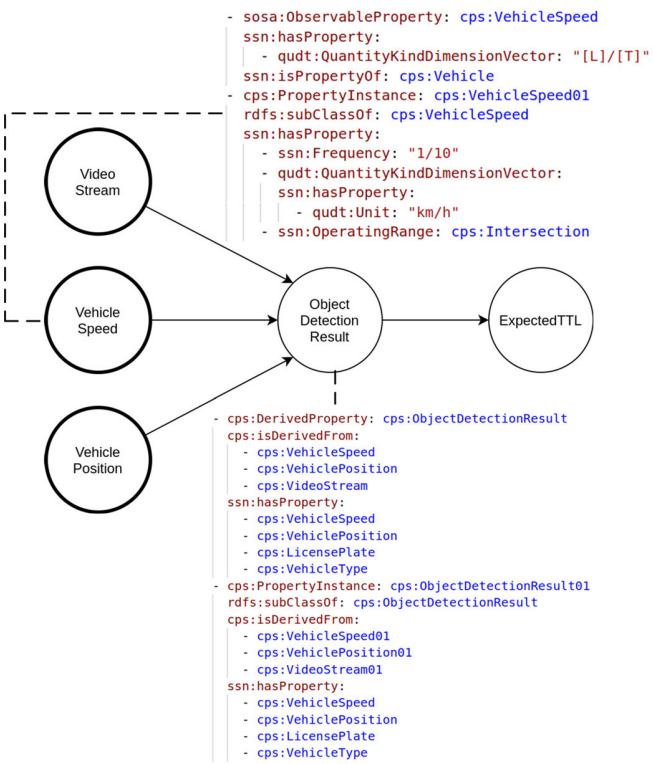


Fig. 2. Semantic specification with the DSL

As a side note, a data stream transformation graph is evidently a dual representation to classic data flow networks. In general terms, edges carry data and nodes perform processing. Similar computational specifications are commonplace in big data processing for apparent reasons. For example directed acyclic graphs of Resilient Distributed Dataset (RDD) transformations in Apache Spark.

Application specifications and requirements formulated as dependency graphs of data streams. Component – importantly: sensor – capabilities are captured in a human-readable, textual domain-specific language (DSL). The DSL heavily relies on semantically standardized concepts and relationships of measurements and observations, taken from the joint W3C and OGC SOSA/SSN (Semantics of Sensors, Observations, Actuation, and Sampling) ontologies [2]. Figure 2 provides a simple example.

Relying on a mapping to the standardized ontologies, the DSL models are translated to a knowledge graph database. The knowledge graph database is the central model store. In which the function mapping of the application specification to sensors, computational blocks and services is planned to be performed (with deployment decisions).

The refined and deployment-extended semantic model is exported to the OASIS standard TOSCA (Topology and Orchestration Specification for Cloud Applications) format. *Software assets* for the realization of the CPS applications are automatically generated and bound as deployable artifacts to the TOSCA template.

This stage enables significant agility in technological realization. In addition ensures conformance with extra-functional requirements through different code generators for data stream nodes and transformation computations.

III. Related Work

As edge-CPS systems have become more complex, there has been a growing need to use supporting tools and technologies. On the one hand, there can be found development and operation supporting frameworks. On the other hand, semantic techniques are also beginning to spread to formulate clear and precise requirements and abilities and to use reasoning techniques. There is also a solution that provides specifically semantic platform support for model-based system design of CPSs [3]. Several tools try to bring together existing technologies, modeling paradigms and integrate them into a toolchain. One such solution is INTO-CPS [4], which provides full lifecycle management of CPS with multidisciplinary modeling capabilities. The tool also approaches modeling in a semantic way for interoperability and verifiability.

In the case of CPS systems, the semantic labeling of the data is also increasingly used. For example, there is a solution for real-time semantic annotation of data, that allows dynamic integration of data on the web [5]. In the context of Industry 4.0, semantic solutions are also used to create intelligent factories by semantic integration of heterogeneous industrial assets [6].

In our solution, we use deployment models, which can be used for many purposes. One possible use is to generate deployments (application codes, configurations, etc.) from the model. TOSCA Lightning [7] is an open-source solution built on the Winery tool. It allows us to visualize, edit, and generate deployment files for deployment models.

IV. Toolchain architecture

The toolchain has a largely sequential architecture and has five major stages. Figure 3 provides an overview.

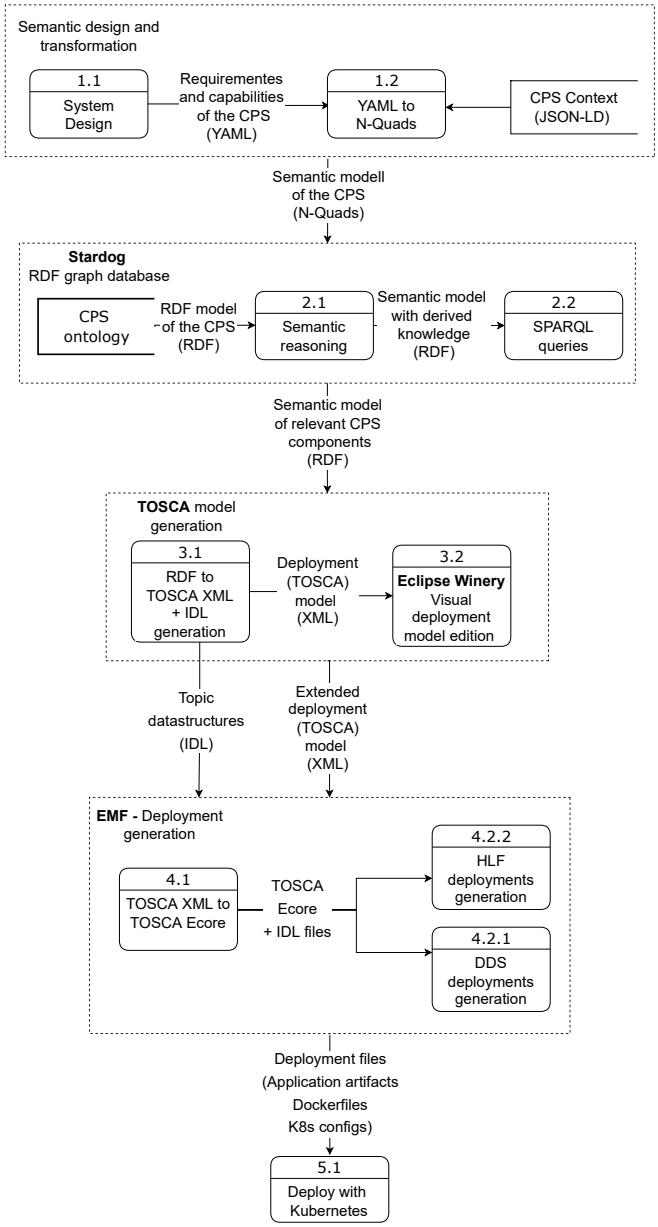


Fig. 3. Data flow diagram of the toolchain

A. Semantic design specification

As a first step, we add semantic knowledge to the data stream representation of our system. In this semantic system description, we use terms and definitions from standard ontologies (such as SOSA, SSN [8]). Many concepts were not available from the existing ontologies, so we had to introduce our own concepts into the *cps* namespace we created. Thereby an additional challenge was the adaptation of our own concepts to the existing ontologies. We also made sure that the combined *cps ontology* covered all the needs to define edge-cps systems (e.g. the definition of requirements and capabilites, procedures, HW-SW components). Some examples of self-defined concepts and expressions used from standard ontologies are shown in Figure 2. Such a self-concept is the *cps:DerivedProperty*, which can be used to describe complex properties (for example can be describe a derivation from the standard *sosa:ObservableProperty* concept). This definition is written in YAML format, which we chose, among other things, for easy human interpretation. This is followed by a combined transformation step, where the YAML format is converted to N-Quads [9].

B. Design refinement in knowledge database

Once we have a semantic description of the system, the next step is to import the description into a graph database platform (Stardog [10]). In the Stardog database, we can extract more knowledge from our system through queries (SPARQL [11]) and reasoning techniques. Using reasoning techniques we can gain extra knowledge about the system. With such additional information, we can answer e.g. the following questions. Is the existing system able to perform the planned tasks? Does the system have redundancy for the specific component? The output of that step is an RDF [12] XML descriptor of the expanded system definition.

C. TOSCA model generation

The next step is to convert the expanded semantic system description to a TOSCA standard deployment descriptor. That deployment model contains the HW-SW nodes and the relationships between them. In our solution, we implemented a transformation tool, which transforms the RDF model into a TOSCA model. The transformation tool uses the Apache Jena [13] Java framework to access the semantic data and map to TOSCA. During the mapping, we also generate IDL [14] files for each data topic of the system. The IDL file generation is not a trivial problem, because there is no standard mapping between the semantic ontology and the IDL files. We had to define the mapping rules (e.g. longitude will be mapped to float *IDL* data type).

After the mapping, the deployment model can be edited visually using the Eclipse Winery [15] tool. In addition to displaying the model, we can make changes, such as which components are placed in which physical computing units.

D. Deployment artifact generation

The next step is code generation. We use EMF (Eclipse Modeling Framework) [16] to map the deployment model to different target technologies. Currently, two platform approaches are supported, both relying on realizing transformations in Docker containers, managed as Kubernetes pods.

- *Logical realization of data stream nodes*: the containers communicate directly, using the RTI implementation of the OMG Data Distribution Services (DDS) [17] specification. DDS ensures the real-time communication between SW components, and meets the unique needs

with the QoS (Quality of Service) options. The Interface Description Language (IDL) files for DDS are automatically generated from the semantic descriptions and the supporting software modules are placed in the generated containers. The logic hosted by the containers is either user-supplied or, for simple transformations, generated automatically.

- *Distributed ledger-based "store and forward"*: for use cases where the Age of Information (AoI) deficiencies of contemporary blockchain platforms (i.e., no hard upper bound) [18] are acceptable, but high resilience and multiparty verification of data are needed, a Hyperledger Fabric [19] distributed ledger deployment is created and the supporting data-handling smart contracts automatically generated. The generated publisher and subscriber containers include the code for writing stream elements to and reading from, the distributed ledger.

E. Deployment

At the end of the toolchain, directly deployable Kubernetes applications emerge where the deployment of containers to the appropriate physical runtime environments (in-field or edge) is enforced by Kubernetes – these constraints were established in the semantic planning phase and also captured in the generated TOSCA template. Note that in contrast to the original intention of the standard, TOSCA here is not used directly for deployment orchestration, as Kubernetes does not have the appropriate support for that yet.

V. Summary and future work

In this paper, we presented a design approach and toolchain prototype for the semantically enabled design and realization of a subclass of CPSs. By utilizing existing standard ontologies for specifying the data streaming through a CPS application, we demonstrated that even mixed-technology realizations can be efficiently created through intelligent code and deployment artifact generation.

Utilizing the semantic inference possibilities facilitated by using a knowledge graph database at the core of our toolchain will be the next step in our research. One major direction is semantic sensor ensemble adaptation for application reconfiguration; initial inroads towards this goal were made in [20] using Formal Concept Analysis (FCA) [21]. The work reported in this paper created a proper technological foundation for this research. The possibilities of using semantic technologies are very wide, our research can even cover the semantic formulation of the behavior of the components. In addition, another relevant research topic is the semantic description of blockchain-based data streams to standardize different blockchain solutions.

References

[1] M. Sabou, S. Biffl, A. Einfalt, L. Krammer, W. Kastner, and F. Ekaputra, "Semantics for cyber-physical systems: A cross-domain perspective," *Semantic Web*, vol. 11, pp. 1–10, 12 2019.

[2] A. Haller, K. Janowicz, S. J. D. Cox, M. Lefrançois, K. Taylor, D. L. Phuoc, J. Lieberman, R. García-Castro, R. Atkinson, and C. Stadler, "The SOSA/SSN Ontology: A Joint W3C and OGC Standard Specifying the Semantics of Sensors, Observations, Actuation, and Sampling," *Semantic Web*, vol. 1, pp. 1–19, 2018.

[3] P. Delgoshaei, M. A. Austin, and A. J. Pertzborn, "A semantic framework for modeling and simulation of cyber-physical systems — nist," 2014.

[4] P. G. Larsen, J. Fitzgerald, J. Woodcock, P. Fritzson, J. Brauer, C. Kleijn, T. Lecomte, M. Pfeil, O. Green, S. Basagiannis, and A. Sadovykh, "Integrated tool chain for model-based design of cyber-physical systems: The into-cps project," in *2016 2nd International Workshop on Modelling, Analysis, and Control of Complex CPS (CPS Data)*, 2016, pp. 1–6.

[5] S. Kolozali, M. Bermúdez-Edo, D. Puschmann, and P. Barnaghi, "A knowledge-based approach for real-time iot data stream annotation and processing," 03 2014.

[6] G. Fenza, M. Gallo, V. Loia, D. Marino, F. Orciuoli, and A. Volpe, "Semantic cpps in industry 4.0," *Advances in Intelligent Systems and Computing*, p. 1057–1068, 2020. [Online]. Available: http://dx.doi.org/10.1007/978-3-030-44041-1_91

[7] M. Wurster, U. Breitenbücher, L. Harzenetter, F. Leymann, and J. Soldani, "Tosca lightning: An integrated toolchain for transforming tosca light into production-ready deployment technologies," in *CAiSE Forum*, 2020.

[8] K. Taylor, K. Janowicz, D. L. Phuoc, A. Haller, M. Lefrançois, and S. Cox, "Semantic sensor network ontology," W3C, W3C Recommendation, Oct. 2017, https://www.w3.org/TR/2017/REC-vocab-ssn-20171019/.

[9] G. Carothers, "RDF 1.1 n-quads," W3C, W3C Recommendation, Feb. 2014, https://www.w3.org/TR/2014/REC-n-quads-20140225/.

[10] "Stardog 7.8.0 documentation," Web Page, accessed: 2021-12-08. [Online]. Available: https://docs.stardog.com/

[11] A. Seaborne and E. Prud'hommeaux, "SPARQL query language for RDF," W3C, W3C Recommendation, Jan. 2008, https://www.w3.org/TR/2008/REC-rdf-sparql-query-20080115/.

[12] P. Patel-Schneider and P. Hayes, "RDF 1.1 semantics," W3C, W3C Recommendation, Feb. 2014, https://www.w3.org/TR/2014/REC-rdf11-mt-20140225/.

[13] "Apache jena webpage," Web Page, accessed: 2021-12-08. [Online]. Available: https://jena.apache.org/

[14] "Idl (interface definition language) specification 4.2," Web Page, accessed: 2021-12-08. [Online]. Available: https://www.omg.org/spec/IDL/4.2/About-IDL/

[15] "Eclipse winery webpage," Web Page, accessed: 2021-12-08. [Online]. Available: https://projects.eclipse.org/projects/soa.winery

[16] "Eclipse modeling framework webpage," Web Page, accessed: 2021-12-08. [Online]. Available: https://www.eclipse.org/modeling/emf/

[17] "Omg data distribution service (dds) 1.4 documentation," Web Page, accessed: 2021-12-08. [Online]. Available: https://www.omg.org/spec/DDS/1.4/PDF

[18] M. Kim, S. Lee, C. Park, and J. Lee, "Age of information analysis in hyperledger fabric blockchain-enabled monitoring networks," in *ICC 2021 - IEEE International Conference on Communications*, 2021, pp. 1–6.

[19] E. Androulaki, A. Barger, V. Bortnikov, C. Cachin, K. Christidis, A. De Caro, D. Enyeart, C. Ferris, G. Laventman, Y. Manevich, S. Muralidharan, C. Murthy, B. Nguyen, M. Sethi, G. Singh, K. Smith, A. Sorniotti, C. Stathakopoulou, M. Vukolić, S. W. Cocco, and J. Yellick, "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," in *Proceedings of the Thirteenth EuroSys Conference*, ser. EuroSys '18. New York, NY, USA: ACM, 2018, pp. 30:1–30:15. [Online]. Available: http://doi.acm.org/10.1145/3190508.3190538

[20] N. Lengyel, R. Szabó, and J. Szalontai, "Edge-alapú kritikus kiberfizikai rendszerek szemantikusan támogatott modellvezérelt telepítése," 2020. [Online]. Available: https://tdk.bme.hu/VIK/info2/alma3

[21] P. Cimiano, A. Hotho, G. Stumme, and J. Tane, "Conceptual Knowledge Processing with Formal Concept Analysis and Ontologies," in *Concept Lattices*, ser. Lecture Notes in Computer Science, P. Eklund, Ed. Springer Berlin Heidelberg, 2004, pp. 189–207.

The Conceptual Framework of a Privacy-aware Federated Data Collecting and Learning System

Levente Alekszejenkó, Tadeusz Dobrowiecki
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: {alelevente, dobrowiecki}@mit.bme.hu

***Abstract*—The federated learning methods offer a strong background of fusing and publishing simultaneously collected data. One of the most challenging problems in federated learning is to hide the identity of the participants. Privacy-preserving techniques try to perturb the participants' local data to match its distribution to the global data.**

In this paper, we consider that agents collect local environmental data. Neighboring agents can share some of their raw data to support real-time decisions and reduce deviation from the global data distribution. The agents will fuse their collected data into a global model that supports the long-term decision and plan making.

We assume the specific situation where the necessary communication protocol between the agents may lead to sharing too much local raw data uncovering private and sensitive attributes of the data sharers.

To handle privacy issues, we introduce a privacy-aware framework. Within this framework, local participants balance the amount of the shared raw data to make it informative enough yet not revealing, effectively bounding the loss of privacy. In this study, we use autonomous vehicle agents as an example to demonstrate the concepts of the proposed framework.

***Index Terms*—privacy-aware, federated learning, autonomous vehicles, data collector systems**

I. Introduction

In recent years, federated learning (FL) methods have made it possible to individually collect non-independent-and-identically-distributed (non-iid) data and to fuse them into a global model effectively. In this paper, we will call the individual participants of the scheme *agents*, and we assume that the global model provides a good approximation of the ground truth of the measured phenomenon. Despite their efficiency, [1] also demonstrated that basic FL methods are easy to break, which raises privacy concerns.

The non-iid data of the agents is the fundamental source of privacy loss. Although there are numerous ways to preserve privacy (see Section III), in this paper, we consider a solution that bounds privacy loss of the agents by sharing some of their raw data. This data sharing will reduce the differences between the data of individual agents, making them indistinguishable from each other.

The research presented in this paper was supported by the Ministry of Innovation and the National Research, Development and Innovation Office within the framework of the Artificial Intelligence National Laboratory Programme.

In this paper, we the propose a horizontally partitioned FL scheme that uses a client-server architecture. The key of the scheme can be summarized as the following iterative steps (see also Fig. 1.):

1) Agents measure a particular phenomenon with their own sensors.
2) Agents identify a subset of agents trusted to share raw data.
3) Agents select a subset of their collected (either by their own sensors or previously received from other agents) raw data to share.
4) Agents share the selected portion of their raw data.
5) Agents train their models with the collected data and update the global model.

We assume that agents in step 3) can compute and limit their loss of privacy. Hence, this pseudo-algorithm is a privacy-aware generalization of the traditional FL methods. The two most interesting parts of the framework are to select the trusted agents and to select the subset of data to be shared. We can *heuristically* choose agents to trust. For example, we trust our partners when text messaging, or our friends when we share photos on social media. To solve the second problem, we assume that the fundamental distribution of the specific problem is known. Hence, each agent can sample its dataset according to this distribution.

Consequently, both questions depend on the specific problem that the FL system has to solve. In Section II, we will give an example of autonomous vehicles (AVs) collecting traffic information. Section III presents a brief overview of the related literature. Section IV describes the rationale of the proposed FL scheme.

II. Privacy-aware Data Fusion of Autonomous Vehicles

Let us assume that AVs collect data about the traffic situation, e.g. parking lot occupation. It seems beneficial to build a predictive model about free parking lots. This model can be requested also by vehicles coming from outside of the measurement area, see Fig. 2 To create such a model, AVs are willing to share their knowledge. These records naturally contain metadata with timing and location information which is enough to identify the route of a particular AV. As it can reveal numerous facts about our daily routine, it is crucially

DOI 10.3311/MINISY2022-013

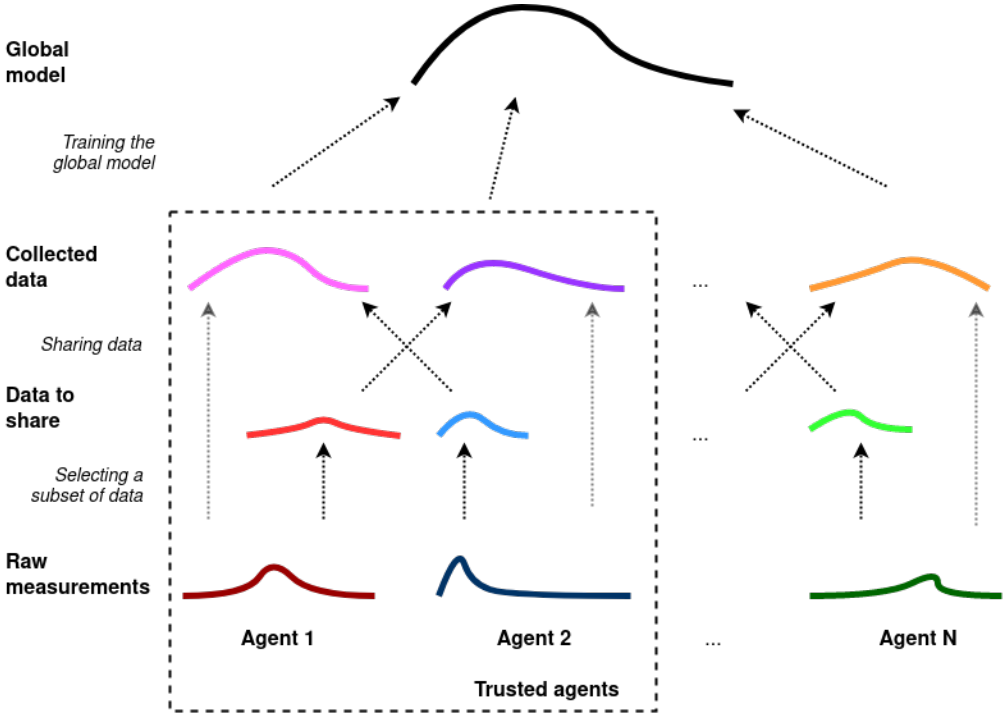


Fig. 1: Overview of the raw data sharing based privacy-aware federated learning system.

privacy sensitive. Consequently, our goal is to hide individual routes as much as possible. To this end, we will apply the framework described in Section I.

Therefore, the agents will be the AVs. For convenience, we will call two interacting AVs as *Ego*, and *Alter*. At first, let us consider when Ego and Alter can trust each other in step 2) of the algorithm. Nowadays, we do not find it a privacy concern to see another car on the streets. This observation can also hold for AVs; therefore, Ego and Alter shall consider each other trusted when seeing each other (being within communication range).

Let us consider two examples to illustrate a possible solution of step 3):

1) The passenger of the Ego vehicle has visited his general practitioner in a neighborhood with sparse traffic and is approaching an arterial street with many cars passing through it. At the corner of this street, Ego meets Alter. If Ego shared the last three minutes of its collected data, Alter would likely infer that it had come from the office of the general practitioner. Therefore, Ego may decide to share only the last 30 s of its recording with Alter.
2) Ego has been going along an arterial street for 5 minutes when Alter comes from the opposite direction. As there are plenty of cars, the last 5 minutes of metadata of Ego is just a representation of the macroscopic traffic flow. Consequently, Ego may share all of its data from the last 5 minutes without any loss of privacy.

III. Literature Review

The communication capabilities of AVs will help the decision-making of the vehicles. Consequently, AVs would enhance the traffic flow, reduce fuel consumption, and mitigate harmful emissions. The key to these improvements is measuring and sharing information about the traffic infrastructure, e.g. about free parking lots [2].

However, the communication of this vast amount of data would require significant bandwidth and processing capabilities. Some solutions require infrastructural elements (e.g. trusted servers, roadside units) to provide privacy-preserving traffic management information [3], [4].

In recent years, FL methods have seemed to solve this technical challenge, even in the field of vehicular networks [5]. The data shared by AVs is crucially sensitive as it can reveal our origins and destinations, even along with timing information; hence, our daily routine. Therefore, it is critical to respect the privacy of the passengers.

There are various methods of privacy-preserving federated learning (PPFL) [6]. Cryptographic techniques are usually computation or communication intensive. Moreover, they may be vulnerable to dishonest servers or agents. Another approach is to use perturbation-based solutions. However, they tend to degrade the model performance. The anonymization-based PPFL, compared to our solution, loses the possibility of utilizing the shared raw data in real-time applications.

The survey of [7] on cyber threats of AVs even calls attention to that vehicles have a relatively long lifespan, and there is no guarantee that cryptographic algorithms and the

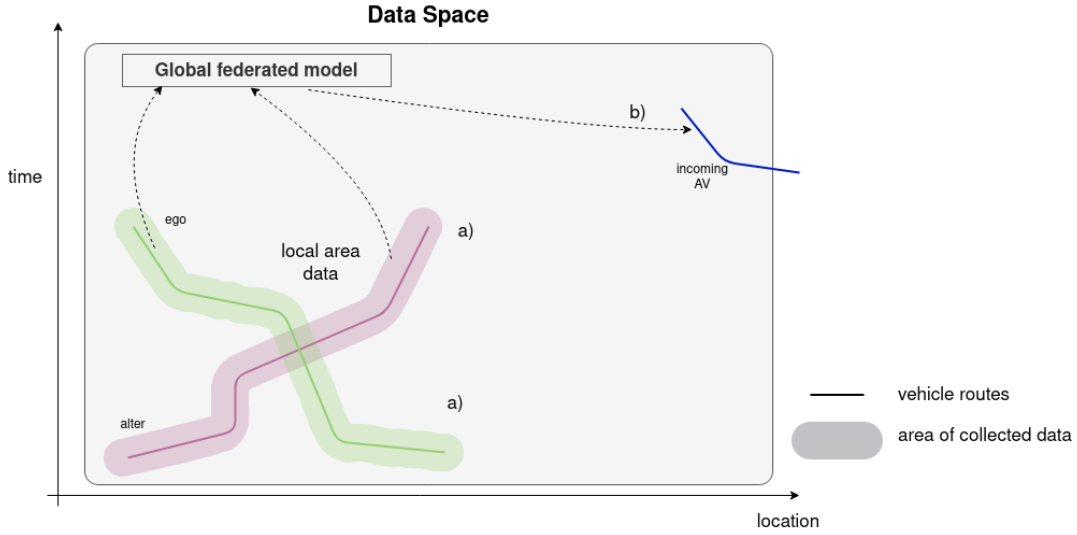


Fig. 2: Measurements have dimensions both in terms of time and space. AVs are sharing some raw measurement data on the Vehicular Ad-hoc Network (VANET). The raw AV sensor data, measured along a specific route (solid lines), will be enriched by information received from nearby vehicles (transparent areas). Hence, their training data will be less divergent to the global model. AVs can update the federated global model a), and they can also request the knowledge of the global model b).

vehicle infrastructure will provide sufficient safety in the future. Reference [8] presents a blockchain-based technique in which agents can share their data by trusted transitions to query and predict the behavior of the system.

The PPFL scheme presented in this paper is a novel, statistics-based solution that aims to minimize the loss of privacy by sharing raw data. Data sharing helps to reduce differences between the knowledge base of the agents. As a consequence, that makes them statistically indistinguishable from each other.

IV. Adversary Models and Defense Methods

Participants of our conceptual framework are the AV agents and a central server responsible for maintaining the global model of the FL method. We assume that each of these participants is honest but curious. As it is required to share information among neighboring agents together with training the global model, we shall consider two types of adversaries: remote and local. Both adversary types intend to perform localization and tracking attacks [9].

A. Remote Adversary

A remote adversary can monitor each AV's modification of the global model. Assuming that a vehicle is likely to move along a specific route and within a specific time span (e.g. commuting to work in the morning), it has a large amount of data about a particular part of a city. Consequently, within each training iteration, it will cause a significant gradient on the corresponding part of the global model. This gradient deviation is likely to reveal the route of the AV, making possible tracking and localization attacks.

The key to solving this problem is blending the revealing gradient around the vehicles' route. Reducing precision can help to achieve this [9], e.g. by introducing geographical zones similar to the TLC Trip Record [10]. However, it also reduces the precision of the global model.

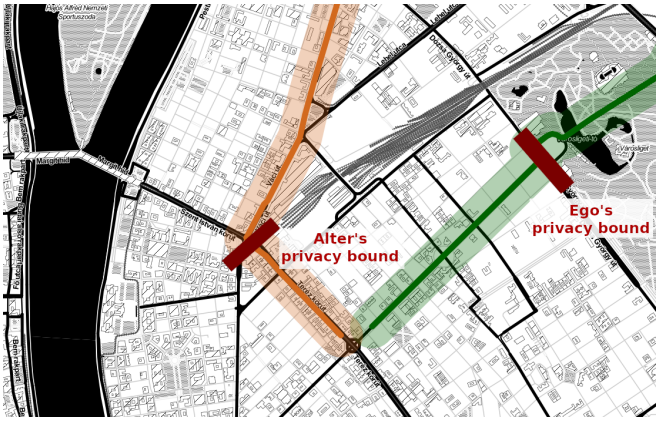
Another approach is to share raw measurement data upon the meetings of AVs. In this way, the data collected by each AV will be a mixture of direct measurements and data received by communication. When there are enough vehicles on the streets, the geo-temporal features of the collected data will be statistically identical to the theoretical traffic flow. Consequently, individual gradients will be homogenous during the training of the global model, making it impossible to reveal the route of a particular vehicle.

Additionally, AVs will also possess real-time information about their environment. This piece of information can also be utilized instantly in decision-making.

B. Local Adversary

Albeit it presents various advantages, sharing raw data among neighboring agents causes a new threat: revealing our route to a possibly adversarial local communication partner. Let us consider that the Ego vehicle meets a malicious Alter. The protocol would dictate that they shall share their current knowledge of the environment. In situations, when Ego has not met enough vehicles, Alter can estimate its route back to the origin. Therefore, a local adversary may be successful in localization and tracking attacks.

On the other hand, the rendezvous itself carries location and time information. Alter can gather this piece of information without sharing any additional data. Due to traffic rules, and depending on the concrete infrastructure, Alter can easily



(a) The AVs (Ego indicated as green, Alter indicated as orange) will only sacrifice a bounded part of their privacy in order to collect information about neighboring streets. Actual routes are marked by solid lines. Before sharing some raw data, the AVs' knowledge bases contain data about the colored areas around their route.

(b) Data shared by Ego (green), and Alter (orange) upon meeting. Both Alter's and Ego's knowledge base will contain data from the indicated areas.

Fig. 3: Privacy-aware local raw data sharing among two AVs (Ego and Alter) when meeting at the Octagonal intersection, depicted at the bottom of the figures.

calculate[1] the statistically most likely previous positions of Ego. If Ego actually does come from this direction, it can share its collected data with minimal loss of privacy.

Moreover, Ego has the advantage that it can exactly compute its loss of privacy when sharing its data with Alter. It can even select a subset of its collected data to minimize this privacy loss. Consequently, Ego can bound its loss of privacy, see Fig. 3.

Considering that sharing too little information among rendezvousing vehicles poses a privacy threat when training the global model, and sharing too much information can be exploited by a local adversary, we expect to find an optimal subset of raw data to share. Therefore, the proposed framework minimizes the expected privacy loss without degrading the precision of the collected data. As it is achieved at the cost of sacrificing a level of privacy, the method is not entirely privacy-preserving, but the privacy loss can be bounded.

[1]For example, the calculation can be based on known traffic volume measures.

V. Conclusion and Further Research Aims

The presented framework is a generalized FL method ensuring a bounded privacy loss of the agents. The scheme includes a step that requires raw data sharing among neighboring or somehow trusted agents. As the framework is currently developed at a conceptual level, our future work involves its implementation and evaluation.

Our current research aims to work out the privacy loss bounded local raw data sharing. We will also investigate the convergence of the planned FL method. As the global model will describe a dynamically changing environment, fast convergence is critical. Additionally, we want to optimize the amount of local data sharing in such a way that it will minimize the overall loss of privacy.

To evaluate the framework, we plan to use simulations. As the scheme requires vehicles meeting on the street, we will test various amounts of traffic (e.g. night, midday, and peak hours). Moreover, different road networks can also influence the working of the scheme. We will carry out simulations on different road topologies: grid-like networks, networks with avenues and boulevards, networks containing bottlenecks.

If evaluations prove the proposed privacy-aware FL scheme beneficial, it can provide a method that guarantees that learning agents can minimize their privacy loss. Within this scheme, the only source of privacy loss is the local data sharing between trusted agents.

References

[1] J. Geiping, H. Bauermeister, and H. Dröge, "Inverting Gradients – How easy is it to break privacy in federated learning?", arXiv:2003.14053, Sep. 2020.
[2] M. Caliskan, D. Graupner, and M. Mauve, "Decentralized Discovery of Free Parking Places", VANET'06, Los Angeles, USA, Sept. 29, 2006.
[3] K. Rabieh, M. M. E. A. Mahmoud, and M. Younis, "Privacy-Preserving Route Reporting Schemes for Traffic Management Systems", IEEE Transactions on Vehicular Technology, vol. 66, no. 3, pp. 2703-2713, March 2017
[4] M. Li et al. "PROS: A Privacy-Preserving Route-Sharing Service via Vehicular Fog Computing," IEEE Access, vol. 6, pp. 66188-66197, 2018
[5] A. M. Elbir, B. Soner, and S. Coleri, "Federated Learning in Vehicular Networks", arXiv:2006.01412, Sep. 2020.
[6] X. Yin, Y. Zhu, and J. Hu, "A Comprehensive Survey of Privacy-preserving Federated Learning: A Taxonomy, Review, and Future Directions", ACM Computing Surveys vol. 54, no. 6, pp. 1-36, July 2022
[7] S. Parkinson, P. Ward, K. Wilson, and J. Miller, "Cyber Threats Facing Autonomous and Connected Vehicles: Future Challenges." IEEE Transactions on Intelligent Transportation Systems, vol. 18, no. 11, pp. 2898-2915, Nov. 2017
[8] Y. Lu, et al., "Blockchain and Federated Learning for Privacy-Preserved Data Sharing in Industrial IoT," IEEE Transactions on Industrial Informatics, vol. 16, no. 6, pp. 4177-4186, June 2020
[9] R. Shokri, G. Theodorakopoulos, J. Le Boudec and J. Hubaux, "Quantifying Location Privacy," 2011 IEEE Symposium on Security and Privacy, 2011, pp. 247-262
[10] NYC Taxi & Limousine Commission, "TLC Trip Record Data", https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page [accessed on Jan. 7, 2022]

The Effect of Transition Granularity in the Model Checking of Reactive Systems

Péter Szkupien, Vince Molnár
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: szkupien.peter@edu.bme.hu, molnarv@mit.bme.hu

***Abstract*—The Theta model checking framework offers the eXtended Symbolic Transition System (XSTS) formalism as a target language for the transformation of high-level models to verify. In XSTS, multiple symbolic transitions can be defined by imperative and declarative statements. The language is flexible enough to offer a broad variety of expressing the semantics of high-level models (e.g., statecharts). Two extremes are i) encoding every (possibly non-deterministic) atomic behavior of the high-level model into a single transition (big steps with only stable states) or ii) modeling the control flow of the computation of the next state (small steps with transient states). Experience shows that big steps are efficient in reducing the state space but sometimes yield transitions that are too complex to handle. Furthermore, internal non-determinism in "big-step" transitions is hard to back-annotate from a counterexample to the high-level model. We examine the effect of transition granularity on model checking by applying a post-processing step that can split "big-step" transitions.**

***Index Terms*—model checking, big-step, transition system**

I. Introduction

In case of safety-critical systems, presumably correct operation is insufficient – it has to be formally proven. Proving the correctness of such software means examining whether it fulfills certain safety requirements. Formal verification is possible on low-level formalisms, but in the case of complex systems, low-level models are usually unavailable – engineers work with high-level ones, instead.

Bridging the gap between these two different abstraction levels is possible with model transformations. In practice, this means using the semantics of the high-level model to transform it to a lower-level formalism on which formal verification is applicable [1]. This also needs to be done backward when annotating the low-level result of formal verification back to the high-level model. The Gamma framework [2] follows this principle to (among others) formally verify component-based reactive systems. Gamma transforms the given high-level input models (modeled with statecharts) and formal requirements (given as CTL expressions) to the low-level XSTS (eXtended Symbolic Transition System) language, executes the model checking of the XSTS model by Theta [3], and annotates the result back to the original model.

This work was partially supported by the National Research, Development and Innovation Fund of Hungary, financed under the [2019-2.1.1-EUREKA-2019-00001] funding scheme.

Gamma transforms high-level statecharts in a highly compressed way, encoding every *run-to-completion* step into a single XSTS transition. The assumption behind this design decision is that avoiding intermediate states can reduce the size of the state space and therefore increase the performance of model checking. This assumption, however, has not been validated, and our experience shows that large monolithic transitions can cause problems in the underlying logic solvers.

The question of how to represent the semantics of a computation in a high-level model is well-researched in several domains, especially in program semantics [4]. So-called *big-step* semantics relates the execution of certain behaviors to their results directly, while *small-step* semantics expresses the execution as a series of elementary steps leading to the final result. In Gamma, big-step semantics is applicable because the run-to-completion step of a statechart is by construction non-divergent, i.e., it will have a finite execution.

This work has two motivations: i) we experienced that in more complex engineering models, a big-step encoding may yield unmanageably large formulas; ii) if a non-deterministically chosen transition contains further internal non-determinism, the result of model checking (i.e., an execution trace) cannot be fully explained in terms of the chosen transitions because the internal decisions are invisible. Both issues led us to the idea of splitting transitions into smaller parts, increasing the *granularity* of transitions.

In this paper, we examine the effect of transition granularity on model checking, most importantly on its performance. We define a method to split big-step transitions into small(er) steps – the prototype is implemented as a post-processing step in the Gamma framework that splits the original "big-step" output. Our results indicate that using big-step semantics in the Gamma framework is indeed a justified choice in terms of performance and any usage of small-step semantics has to be carefully designed to avoid the large overhead.

II. Background

A. eXtended Symbolic Transition Systems

In this work we describe systems using *eXtended Symbolic Transition Systems* (XSTS) [5]. We define an XSTS model as a 4-tuple $XSTS = \langle V, Tr, In, En \rangle$, where:

- $V = \{v_1, v_2, \ldots, v_n\}$ is a set of *variables* with domains $D_{v_1}, D_{v_2}, \ldots, D_{v_n}$, e.g. *integer*, *bool*, or *enum*;

DOI 10.3311/MINISY2022-014

- A state of the system is $s \in S \subseteq D_{v_1} \times D_{v_2} \times \cdots \times D_{v_n}$, which can be regarded as a value assignment: $s(v) \in D_v$ for every variable $v \in V$. The initial state s_0 is given as the initial value for each variable.
- $Tr \subseteq S \times S$ is the *internal transition relation*, describing the behaviour of the system itself;
- $In \subseteq S \times S$ is the *initial transition relation*, describing the initialization of the system, which is executed only once at the beginning of the execution;
- $En \subseteq S \times S$ is an abstraction of the *environmental transition relation*, describing the environment the system is interacting with;
- Both Tr, In and En may be defined as a union of exclusive transitions that the system can take. Abusing the notation, we will denote these transitions as $t \in Tr$ which actually means that $t \subseteq S \times S$ as a transition relation is a subset of Tr.

From the initial state s_0, In is executed exactly once. Then, En and Tr are executed in alternation. In state s, the execution of a transition relation T (being either of the transition relations) means the execution of exactly one non-deterministically selected $t \in T$ transition. In addition to the non-deterministic selection, transitions may be non-deterministic internally, therefore $t(s) = \{s'_1, \ldots, s'_k\}$ yields a set of successor states.

Transitions are described as $op \in Ops$ *operations*, which may be composite operations. The semantics of transitions are defined through the semantics of operations, which is, in turn, the definition of op as a relation over $S \times S$. For a precise description, refer to [5] – for this work, an informal definition is sufficient. XSTS defines the following *basic operations*:

- *Assignments*: An assignment of form $v := \varphi$ with $v \in V$ and φ as an expression of the same type means that φ is assigned to v in the *single* successor state s' and all other variables keep their value.
- *Assumptions*: An assumption of form $[\psi]$ with ψ as a Boolean expression checks condition ψ without modifying any variable and can only be executed if ψ evaluates to *true* over the current state s, in which case the single successor state is $s' = s$ – otherwise the set of successor states is the empty set $\emptyset$.
- *Havocs*: A havoc of form $havoc(v)$ with $v \in V$ means a non-deterministic assignment to variable v, i.e., after executing the transition the value of v can be anything from D_v. Therefore, the number of successor states s'_i will be equal to the size of D_v.
- *Local variables*: A local variable can be declared as an operation of form $var\ v_{loc} : type := \varphi$.[1] A local variable can only be accessed in its *scope* which is its direct container composite operation. Technically, the declaration of a local variable v_ℓ adds it to V and assigns its initial value φ to v_ℓ while the end of every scope removes every local variable declared in it from V.

[1]The default value of the type is used as an initializer unless explicitly specified by the modeler.

Composite operations contain other operations but their execution is still atomic. Practically, this means that the contained operations are defined over transient states and the composite operation determines which one(s) will be the (stable) result of the composite operation. XSTS defines the following composite operations:

- *Sequences*: A sequence is composed of operations $op_1, \ldots, op_n$ with $op_i \in Ops$ executed sequentially, each applied on every successor state of the previous one (if any). The set of successor states after executing the sequence is the result of the last operation.
- *Choices*: A choice means a non-deterministic choice between operations (branches) $op_1, \ldots, op_n$ with $op_i \in Ops$. This means that exactly one branch op_i will be executed. The set of successor states is the union of the results of any branch.

Note that assumptions may cause any composite operation to yield an empty set as the set of successor states. This allows us to use the *choice* operation as a guarded branching operator, ruling out branches where an assumption fails by yielding an empty set as the result of that branch.

In this work, we make the following assumptions, which can be easily guaranteed by simple pre-processing. *1)* The operation of transitions and non-deterministic choice branches must be composite actions. Thus, single basic operations will be treated as 1-long sequences. *2)* We assume that there are no sequential actions directly inside sequential actions. These restrictions help the clarity and consistency of local variable scopes without the loss of generality.

B. Formal Requirements

In Gamma, the verifiable requirements towards the system are formalized in a subset of computational tree logic [6] (CTL). When using Theta as a model checker, only two CTL operators are allowed: $AG\phi$ means that ϕ should be true in every reachable state, while $EF\phi$ means that there should be at least one reachable state satisfying ϕ. The Boolean expression ϕ is defined over the variables or states of the model.

III. Splitting Transitions

Given an XSTS model, our goal is to split its monolithic transition(s) into smaller transitions (called *fragments*) without modifying its semantics. Splitting is a *model transformation* over XSTS models yielding another XSTS model $\langle V', Tr', In', En' \rangle$. Informally, this means selecting some transitions containing composite operations and splitting them by putting some of their contained operations into separate transitions (so $|T'| \geq |T|$ always holds for any transition relation that has a split transition). With this approach, we have four main challenges:

- *Granularity*: We need to decide which composite actions to select for splitting, and which of their contained operations to put into separate transitions.
- *Control flow*: Composite operations have their own semantics which we must maintain. Splitting a composite

operation means that it will possibly be completely removed from the original model and only its contained operations will be kept as separate transitions. Additional variables and operations may be necessary to emulate the effect of the original composite operation.

- *Atomicity*: Splitting a composite action into separate transitions breaks the atomicity of the original composite action fundamentally modifying the semantics of the model while adding transient states to the system which are unobservable in the original model.
- *Local variables*: Splitting can also cause the declaration and usage of a local variable to end up in different transitions (fragments). Therefore, some local variables may become global and management of the scope has to be emulated with additional operations.

A. Granularity

One of our motivations in this work is to eliminate internal non-determinism, therefore we have opted for splitting by non-deterministic *choice* operations. The goal is to have every branch of *choice* operations as a separate transition. Due to the composite nature of operations, this implies a transformation for *sequences* as well: every *sequence* has to be split by every *choice* it contains. Consequently, a *sequence* with n *choices* will be split into at most $n+1$ fragments plus the fragments coming from the *choice* – empty fragments are discarded.
Without deeper analysis, we also point out a few other options.

- Putting every basic operation into a separate transition is the finest possible granularity. Such models will have no internal non-determinism other than *havoc* operations.
- Since *havoc* operations also introduce non-determinism, it would be reasonable to put them into separate fragments and treat them specially.
- Very long deterministic *sequences* may also be split by considering a parameter k as the maximum length of fragments. In this case, an n-long *sequence* would be split into $\lceil n/k \rceil$ fragments.

B. Control Flow

To maintain the semantics of composite actions we introduce a new integer variable l ($D_l = integer$) to the system ($V' = V \cup \{l\}$) as a *program counter* to emulate the control flow of the original composite operations (the default name is `__pc` and it can be reused for all the transition relations).

This program counter l stores the state of execution for the original operation in transient states and we enclose every transition in between an assumption and an assignment reading and updating its value. The assumption works as a *guard* of the transition enabling its execution only if it could execute according to the original semantics. The assignment updates l after the successful execution of the fragment, enabling other fragments or leading back to a stable state.

Since the value of l is modified only at the end of fragments, its values can be regarded as a labeling of fragments. We use the notation $l(t)$ for the value of l inside fragment t. The first fragment(s) begin(s) with an assumption of $[l = 0]$, and the last fragment(s) end(s) with an assignment of $l := 0$.[2] Similarly, first/last fragment(s) of *choice* branches assume/assign the same label to split/merge the control flow, respectively.

An example XSTS code snippet and its splitting are shown below. The original model has variables $V = \{a, i\}$ and a single transition $Tr = \{t\}$ consisting of an *assumption*, a *local variable declaration*, a *choice* (with two branches) and *assignments*. The split model has variables $V' = V \cup \{x\} \cup \{l\}$ where x was a local variable made global and l is the program counter (`__pc`). $|Tr'| = 4$ because the original transition is split into a 2-long and a 1-long sequence before and after the choice which has 2 branches.

Original XSTS

```
var a, i : int;
trans {
  assume i<10;
  var x : int = 1;
  choice {
    a := a+1;
    x := x+1;
  } or {
    a := a+2;
    x := x+2;
  }
  i:=i+x;
}
```

Split XSTS

```
var a, i, x, __pc : int;
trans {
  assume __pc == 0;
  assume i<10;
  x := 1;
  __pc := 1;
} or {
  assume __pc == 1;
  a := a+1;
  x := x+1;
  __pc := 2;
} or {
  assume __pc == 1;
  a := a+2;
  x := x+2;
  __pc := 2;
} or {
  assume __pc == 2;
  i := i+x;
  x := 0;
  __pc := 0;
}
```

C. Atomicity

Even with the same control flow, transient states modify the semantics of a split model with regard to CTL formulas. To compensate for this, we need to restrict the formula to apply only on *stable* states – these are exactly the ones where l is 0. Depending on the temporal operator, the following transformation is applied:

- $AG\phi \rightsquigarrow AG(l = 0 \implies \phi)$, i.e., ϕ is evaluated only in *stable* states;
- $EF\phi \rightsquigarrow EF(l = 0 \wedge \phi)$, i.e., there is a *stable* state satisfying ϕ.

D. Local Variables

After the splitting of a transition relation T, a post-process step is needed to fix the broken local variables whose declaration and usage ended up in different fragments. $U \subseteq V_\ell \times T'$ is the usage relation of the local variables, where V_ℓ is the set of local variables declared in any $t \in T$, and T' is the transition relation after splitting. We have $(v_\ell, t) \in U$ *iff* transition t uses local variable v_ℓ either in i) a local variable declaration, ii) an assignment $v_\ell := \varphi$, iii) an expression φ (either in an assumption or the right-hand-side of an assignment or

[2] Multiple first/last fragments can occur if *sequences* start/end with *choices*.

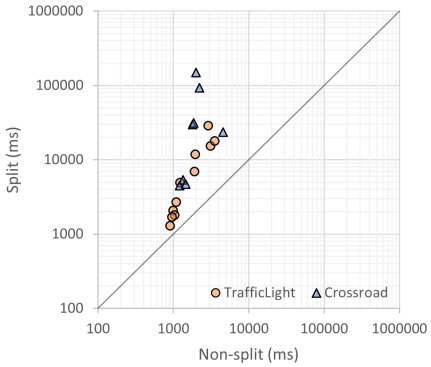


Fig. 1. The time of verification with and without splitting.

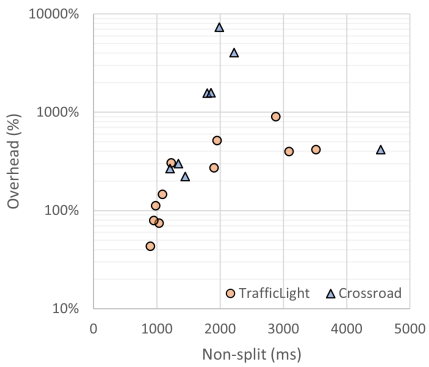


Fig. 2. The overhead of splitting by verification time (without splitting).

declaration), or iv) a havoc $havoc(v_\ell)$. For every $v_\ell \in V_\ell$, if $(v_\ell, t_1), (v_\ell, t_2) \in U$, $t_1 \neq t_2$, we execute the following:

1) Make the local variable *global* by removing the local declaration and adding v_ℓ to V'. The original initial value φ of v_ℓ is replaced with the default value $d \in D_{v_\ell}$.
2) Add an assignment $v_\ell := \varphi$ to the original place of the local variable declaration (if φ is not the default value).
3) Append an assignment $v_\ell := d$ to every end fragment of the scope of v_ℓ to reset the variable. This transformation guarantees that $v_\ell = d$ outside of the original scope of v_ℓ. This is not required by the semantics but helps in reducing the state space.

IV. Evaluation

We conducted measurements of the execution time of model checking with Theta on various XSTS models and formal requirements before and after applying splitting. Every measurement was run 5 times, with the JVM flag `-Xmx6144m` (6 GB heap memory), on a machine with Windows 10, 16 GB RAM and Intel® Core™ i5-10310U CPU. We had the following research questions:

- **RQ1**: Can splitting solve the problem of unmanageably large transitions?
- **RQ2**: How does the performance of a small-step semantics compare to the big-step semantics used by Gamma?

To answer RQ1, we have executed Theta on the split version of an industrial-scale model from [7] with the formal requirement that previously caused Theta to crash on the size of transitions. Although the crash did not occur with splitting, 30 hours were not enough to finish verification.

To answer RQ2 and assess the effects of splitting, we used two simpler example models available as part of Gamma. One of them is the state machine describing a traffic light, while the other one is a composite system including multiple traffic lights to model a crossroad. Multiple requirements have been generated with Gamma to reach certain states or transitions in the models. The results are visualized in two figures.

Fig. 1 shows how the execution time of Theta compares for each case with or without splitting. Splitting always resulted in a higher execution time, often quite drastically. This seems to indicate that the choice of big-step semantics during the design of Gamma is justified. Moreover, if the developers choose to implement a small-step semantics to address the problem of back-annotating internal non-determinism, extreme caution should be taken to avoid the unacceptable overhead we have experienced in some cases (see Fig. 2).

V. Conclusion and Future Work

This paper showed that using a small-step semantics for the transformation of statecharts into low-level verification models can cause a huge overhead. While it is tempting to use a small-step semantics to address the problem of overly complex transitions and the lack of back-annotation for internal non-determinism, the currently implemented big-step semantics perform much better in the analyzed cases.

Nevertheless, small-steps semantics may have other useful applications. Based on the implemented post-processing splitting transformation, we plan to implement i) a simulator for XSTS making every non-deterministic choice explicit to the user (simulators do not suffer from the performance problems typical in model checking) and ii) a hybrid model checking approach that uses small-step semantics only when a counterexample is found, guiding the search in the split model based on the trace obtained from the big-step model (such an approach could combine the benefits of both worlds).

References

[1] E. M. Clarke, T. A. Henzinger, and H. Veith, "Introduction to model checking," in *Handbook of Model Checking*, E. M. Clarke, T. A. Henzinger, H. Veith, and R. Bloem, Eds. Springer Cham, 2018, pp. 1–16.
[2] V. Molnár, B. Graics, A. Vörös, I. Majzik, and D. Varró, "The Gamma statechart composition framework: Design, verification and code generation for component-based reactive systems," in *Proceedings of ICSE'18: Companion Proceeedings*. ACM, 2018, pp. 113–116.
[3] T. Tóth, Á. Hajdu, A. Vörös, Z. Micskei, and I. Majzik, "Theta: A framework for abstraction refinement-based model checking," in *Proceedings of FMCAD'17*, Vienna, Austria, 2017, p. 176–179.
[4] X. Leroy and H. Grall, "Coinductive big-step operational semantics," *Information and Computation*, vol. 207, no. 2, pp. 284–304, 2009.
[5] M. Mondok, "Formal verification of engineering models via extended symbolic transition systems," Bachelor's Thesis, BME, 2020.
[6] E. M. Clarke and E. A. Emerson, "Design and synthesis of synchronization skeletons using branching time temporal logic," in *Logics of Programs*. Berlin, Heidelberg: Springer, 1982, pp. 52–71.
[7] B. Horváth, B. Graics, Á. Hajdu, Z. Micskei, V. Molnár, I. Ráth, L. Andolfato, I. Gomes, and R. Karban, "Model checking as a service: Towards pragmatic hidden formal methods," in *Proceedings of MODELS'20: Companion Proceedings*, 2020.

Towards Hand-over-Face Gesture Detection

Gábor Révy, Dániel Hadházi, Gábor Hullám
Department of Measurement and Information Systems
Budapest University of Technology and Economics
Budapest, Hungary
Email: revy.gabor@edu.bme.hu, {hadhazi, hullam.gabor}@mit.bme.hu

***Abstract*—Facial microexpressions are immediately appearing reactions on the face that indicate various details about people's mental and emotional states. Their most important property is that their interpretation is identical or very similar for people all over the world. At present, their identification requires a psychologist expert. Thus automating this task would enable a broader application.**

The goal of this research is the detection of microexpressions using hybrid expert algorithms. Our algorithms mainly rely on landmark point detectors. Based on their output, several expert algorithms are utilized to extract key features and changes appearing on the face of a subject. These algorithms usually include several steps of image processing and time series analysis algorithms.

In this paper, a component responsible for detecting hand gestures and hand pose is introduced. This component helps other algorithms to eliminate false positive detections by detecting the hands over the face. In addition, the recognizability of hand-over-face gestures is investigated. Finally, the implemented face occlusion detector method is evaluated on videos.

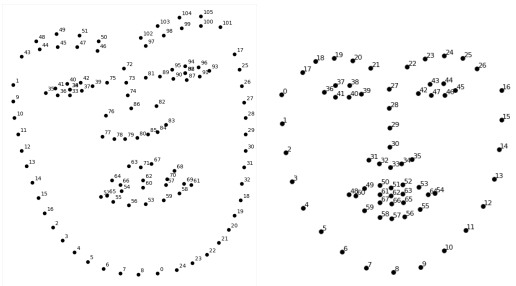
***Index Terms*—microexpression, image processing, landmark points, expert system, facial expressions, hand-over-face gestures**

I. Introduction

Microexpressions are the visible features of emotions appearing on the face for a very short time, e.g. an involuntary reaction to a question. Automating the detection of facial expressions would allow a wide range of uses, e.g. to study reactions to an advertisement or to assist in the diagnosis of mental disorders. Experts usually distinguish 7 different basic emotions: anger, disgust, fear, happiness, sadness, surprise and contempt. In order to categorize reactions in videos, proper detection of microexpressions is required.

The first step is to detect the motion of the muscles, the so-called action units on the face. These parts of the face are described in detail in the FACS system [1]. Emotions can be determined based on the activated action units.

Previously, we have developed several algorithms to identify these muscle movements appearing on the face. These methods operate on videos, since our goal is to detect the change in the facial features. Our algorithms utilize a facial landmark point detector to localize key points in the face. We combine two facial landmark detectors to make the localization more accurate and robust. One is the neural network-based PFLD [2] and the other, that can be found in the Dlib [3] library is utilizing an ensemble of regression trees. These identify 106 and 68 key points on the face, respectively, as shown in Figure 1. Using the landmark detectors on videos also allows for smoothing between the frames. This is useful because the output of the detectors often oscillates, it is not accurate, and some objects can even occlude the face.



(a) PFLD [2] (b) Dlib [3]

Fig. 1: Facial landmark points identified by the facial landmark detectors utilized.

Furthermore, the fact whether the hand is occluding the face, can be an additional source of information to determine emotions [4]. This already has a meaning in itself, but it can also help other image processing algorithms to indicate that there may be anomalies or outliers due to this "noise". Some algorithms already exist to detect hand-over-face (HOF) gestures. Mahmoud et al. utilized histograms of oriented gradients (HOG) and local space-time features [5], [6] to extract features and an SVM (Support Vector Machine) for classification to detect HOF gestures. In addition, Mahmoud et al. [7] compared the use of local binary patterns and Gabor filters in detecting face occlusions. Ghanem et al. constructed a neural network called MPSPNet [8] to segment the hands over the face.

In this paper a simple, but surprisingly accurate approach is proposed for hand over face detection. Furthermore, the pose of the hand may reveal additional information about the mental state of the person in the video [4]. This area however, requires further investigation and it is out of scope of this paper. Here we only focus on the recognizability of different hand poses.

This research was supported by the ÚNKP-21-5-BME-362 New National Excellence Program of the Ministry for Innovation and Technology from the source of the National Research, Development and Innovation Fund, and the János Bolyai research scholarship.

Figure 2 provides an overview of the tasks described in this paper.

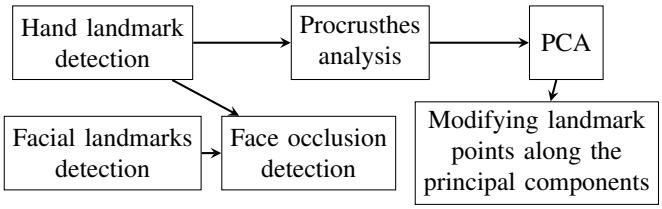


Fig. 2: Steps of the tasks related to hand-over-face gesture detection.

II. Hand over face detection

There are several tasks related to hand-over-face detection (HOF) including face occlusion detection and hand pose detection. After the hand detection, it can be informative for other algorithms to know, whether a specific part of the face is occluded. This is because covering the face acts as noise in algorithms that focus on different areas of the face. In addition, it also carries information about emotions, i.e. the covered area of the face can indicate certain types of emotions. Detecting hand pose along with the position could reveal even more information about emotions.

To detect hands and hand landmarks the MediaPipe [9] library was utilized. MediaPipe is an open source cross-platform library providing customizable machine learning solutions for visual media (i.e. images and videos). It contains several landmark and object detection algorithms as well as segmentation algorithms. MediaPipe Hands is a hand and hand landmark detector. It detects the ROI (region of interest) and 21 3D key points of the hands in an RGB image (see Figure 3). The handedness of the hand ROI is also determined.

Fig. 3: 21 landmark points (and their skeleton) detected by the MediaPipe [9] Hands detector.

A. Face occlusion detection

To detect the occlusion of different areas of the face, the face is divided based on its landmark points. 3 areas were selected on the face, that are separated by the bottom of the nose and the upper line of the eyes. The occlusion is simply determined by checking if a hand landmark point falls into such an area. Examples of the output of the face occlusion algorithm are shown in Figure 4. Note, that depending on the analyzed microexpressions, additional facial regions of interest could be defined.

Fig. 4: The output of the face occlusion detection: the occluded part is marked with red. The source of the original images: BAUM-1s dataset [10]

Using this algorithm jointly with other feature detectors allows the removal of several false positive detections. An example of this is shown in Figure 5: the lip compression detector algorithm made a false positive detection, but using the HOF detector, this can be removed automatically.

B. Towards hand gesture recognition

The detectability of hand gestures was also investigated. A series of hand gestures were recorded along with their landmark points. In order to eliminate the effect of the scaling, rotation and translation of the landmark points and investigate only the relevant information of the shape, the coordinates of the landmark points must be normalized appropriately. This can be performed by the Procrustes superimposition [11]. The Procrustes superimposition algorithm is usually applied before comparing shapes. It minimizes the so-called Procrustes distance between two shapes by scaling, translating

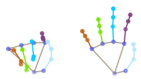
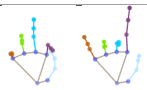
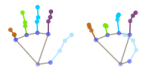
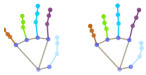
Fig. 5: The lip compression detector made the mistake of a false positive detection, but using the face occlusion detector, it can be filtered out. Source: BAUM dataset [10]

and rotating them. In our case the Procrustes superimposition algorithm minimized the Euclidean distance between the palm landmark points of the recorded hand and a simple hand model. After the uniformization, PCA (Principal Component Analysis) was applied on the coordinates of the landmark points. The motivation of the utilization of PCA is to explore and coordinate the manifold of the landmarks in order to extract features, which can efficiently describe the pose of the hands. Note that PCA extracts features along which the deviation of the samples is the highest. Therefore, we can adapt these features to our goal by expanding the dataset with specific samples.

The most relevant features (principal components) were selected to explain 95% of the variance in the data, i.e. the first 8 components. After projecting to these components and modifying their values, the reconstructed landmark points were investigated. For each principal component, Table I shows the changes observed when examining the restored landmarks of the hand.

As Table I shows, the most relevant principal components can be interpreted semantically fairly well. The detection of the hand gestures is currently in progress. It could be utilized later to look for signs of different mental states. One such sign is nail biting, which is an indicator of stress. Other detectable states include thinking, evaluation, skepticism, boredom, happiness and deception [12]. A further option is to take multiple aspects of hand features into consideration such as hand shape and hand action [4]. Hand shapes range from closed hand through holding out one or more fingers to a fully open hand, while hand action includes holding (the chin), leaning (the face on the hand), tapping and touching (parts of the face). Using these ”dimensions” as descriptors, a *thinking* state could be characterized with using the index finger (or other fingers) to touch the chin or the forehead, whereas a *bored* state would be better characterized by either leaning the face on an open or closed hand or holding the chin. The challenge with this approach however is that there are no properly annotated videos for hand gesture detection. Existing datasets either consist of only still images or they lack the necessary annotation that is detailed enough to be of any use. The BAUM dataset [10] could be utilized for hand gesture detection, however this requires additional annotation

TABLE I: The first 8 principal components explaining 95% of the variance in the recorded hand landmark data.

	Description	**Cumulative explained variance**	**Plot**
1	openness of fingers 2-5	0.5	
2	slant of the openness of fingers 2-5	0.6676	
3	openness of the 2nd and 5th fingers with respect to the 3rd and 4th fingers	0.7714	
4	openness of the thumb	0.8415	
5	curvature of the thumb	0.9003	
6	spreadedness of the fingers	0.9225	
7	distance between the fingers	0.9393	
8	lateral position of the 3rd and 4th fingers	0.954	

and further investigation of applicable expert algorithms.

III. Evaluation of hand occlusion detection

To evaluate the hand-over-face (HOF) detection we were looking for annotated datasets specific to this purpose. However, we only found a dataset containing images labeled with hand segmentation. Unfortunately, this is not fit for purpose, as the algorithm that detects the facial landmark points can only be run on videos to smooth the detection between the frames. This is useful, because if the face is occluded by the hand, the landmark points are not detectable by default, but can be determined from the adjacent frames. Thus, the detection of the HOF gestures was evaluated on videos selected from the same BAUM [10] dataset, which we used previously to evaluate other microexpression detection algorithms. A total of 50 videos were selected in which the hand covers the face. The videos contained 79 HOF gestures from which 76 (96.2%) was detected and there were 3 false positive detections. When examining the results, the following could be observed:

- The detector performs well if a large part of the hand is visible and the hand is not too blurred. If the frame rate is high enough, the image will not become blurry even during rapid hand movements.
- In a very small number of cases, the hand detector generates a false detection of hand. In those cases, where the illusionary hand landmark points do not intersect the area covered by facial landmark points, this is not a problem,

as it does not cause a false occlusion (see Figure 6a). In cases where there is an intersection however, this causes false detection (as shown in Figure 6b).

- If a big part of the hand is not visible or the hand is in an uncommon position (or at least a position difficult for the detector to detect), the hand landmark points become inaccurate. This can lead to false detections (as shown in Figure 6c).
- If the hand moves too fast or the frame rate is too low compared to the speed of the hand movement, the image of the hand becomes blurry and the hand is not detected. This can lead to false negative detections as shown in Figure 6d.
- If only a small part of the hand is occluding the face, no occlusion may be detected as can be seen in Figure 6e. This is caused by the nature of the HOF detection algorithm. This could be addressed by creating a hand segmentation based on the hand landmark points and examining the cross section of the hand and the face mask.

(a) Fake hand landmark detections.

(b) False positive detection caused by a fake hand detection.

(c) False positive detection caused by the inaccuracy of the hand landmark detector.

(d) False negative detection caused by a false negative hand detection.

(e) False negative detection caused by the nature of the hand over face detection algorithm.

Fig. 6: Examples of difficult hand over face detection cases. Source: BAUM dataset [10]

IV. Conclusion

In this paper, we introduced a component to detect the pose of the hand over the face. The utilization of this algorithm eliminated several false positive detections by notifying other algorithms that their ROI is covered by the hands. We also investigated the recognizability of the hand-over-face gestures and have shown, that the principal components defined by applying PCA can be interpreted semantically. Finally, we evaluated the face occlusion detector method on videos from the BAUM [10] dataset, which is specialized for emotions. Results have shown, that despite of being a simple approach, the face occlusion detector works quite accurately. This however requires that the two machine learning-based algorithms, (i.e. the face and the hand landmark detectors) we are building on, work correctly. In the future we plan to extend the algorithm with additional features such as taking hand actions into consideration.

References

[1] P. Ekman, J. C. Hager, and W. V. Friesen, *Facial action coding system: the manual*. Research Nexus, 2002.

[2] X. Guo, S. Li, J. Yu, J. Zhang, J. Ma, L. Ma, W. Liu, and H. Ling, "Pfld: A practical facial landmark detector," *arXiv e-prints*, pp. arXiv–1902, 2019.

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

[4] M. Mahmoud and P. Robinson, "Interpreting hand-over-face gestures," in *International Conference on Affective Computing and Intelligent Interaction*, pp. 248–255, Springer, 2011.

[5] I. Laptev, "On space-time interest points," *International journal of computer vision*, vol. 64, no. 2, pp. 107–123, 2005.

[6] M. Mahmoud, T. Baltrušaitis, and P. Robinson, "Automatic analysis of naturalistic hand-over-face gestures," *ACM Transactions on Interactive Intelligent Systems (TiiS)*, vol. 6, no. 2, pp. 1–18, 2016.

[7] M. Mahmoud, R. El-Kaliouby, and A. Goneid, "Towards communicative face occlusions: machine detection of hand-over-face gestures," in *International Conference Image Analysis and Recognition*, pp. 481–490, Springer, 2009.

[8] S. Ghanem, A. Dillhoff, A. Imran, and V. Athitsos, "Hand over face segmentation using mpspnet," in *Proceedings of the 13th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 1–8, 2020.

[9] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, *et al.*, "Mediapipe: A framework for building perception pipelines," *arXiv e-prints*, pp. arXiv–1906, 2019.

[10] S. Zhalehpour, O. Onder, Z. Akhtar, and C. E. Erdem, "Baum-1: A spontaneous audio-visual face database of affective and mental states," *IEEE Transactions on Affective Computing*, vol. 8, no. 3, pp. 300–313, 2016.

[11] M. Rudemo, "Statistical shape analysis. I. L. Dryden and K. V. Mardia, Wiley, Chichester 1998. no. of pages: xvii+347. ISBN 0-471-95816-6," *Statistics in Medicine*, vol. 19, no. 19, pp. 2716–2717, 2000.

[12] A. Pease and B. Pease, "The definitive book of body language," 2006.

Using Dimension Reduction Methods on the Latent Space of Molecules

György Józsa, Péter Sárközy
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
Email: jozsa.gyuri9910@gmail.com, sarkozy.peter@vik.bme.hu

***Abstract*—*De novo* molecule design is the process of generating novel chemicals based on a dataset of drug-like molecules. This method has gained popularity in recent decades.**

Developing drug-like molecules is both costly and time-consuming. To speed the process up, machine learning and deep neural networks have been used in the last three decades. A particularly popular method is using a variational autoencoder to generate a latent space of drug-like molecules suitable for targeted searching.

Quantifying the quality of such a latent space is vital for effective usage. This task is not trivial however, as the chemical structure of molecules cannot be easily quantized and such latent spaces tend to be high-dimensional, leading to the need for dimension reducing visualization algorithms to be applied.

Many dimension reduction and visualization algorithms have been developed in recent decades. In this paper, we evaluate five recent algorithms – PCA, t-SNE, UMAP, TriMAP and PaCMAP – to see how well they perform on a given dataset.

We examine each algorithm on its ability to transform a 64-dimensional latent space such that the resulting two-dimensional space is smooth over chemical structure. We optimize the hyperparameters of each algorithm to see how they transform the resulting embedding and perform a linear interpolation test to see how they map the latent space into two dimensions. We examine the invertibility and extensibility of each algorithm, as this can make targeted searching much easier to execute.

***Index Terms*—data science, data visualization, dimension reduction, drug discovery, machine learning**

I. Introduction

One of the more important goals of modern pharmaceutical research is the discovery of novel drug-like molecules. This development however is a long and costly process. In order to reduce the time cost of the procedure, as well as the financial cost, targeted search of the chemical space is necessary. Using targeted searching on the space of drug-like molecules is a heavily researched topic.

Algorithms exist for generating suitable molecules, however, finding efficient searching algorithms without enumerating every element of the subspace is difficult. Deep neural networks have been used for *de novo* molecule generation with promising results. One particularly promising method is the use of variational autoencoders to transform the chemical space into a latent space that is smooth over the latent space of chemical structures.

The number of dimensions of an autoencoder's latent space is usually between 32 and 128, which is rather large, and presents a challenge in understanding the underlying structure. The analysis of such latent spaces is important for finding novel molecules with desirable drug-like properties.

II. Algorithms

Principal Component Analysis [1], or PCA, for short, is the most widespread dimension reduction algorithm. It works by finding *principal components*. These are a series of p eigenvectors, where the i-th vector minimizes the average squared distance from the points to the line while also being orthogonal to the first $i-1$ vectors.

PCA is the process of computing the principal components of the data and performing a change of basis, usually using only the first few principal components and ignoring the rest. This leads to a lower dimensional representation of the data, where only the components that preserve as much of the data's variation as possible are included.

t-distributed stochastic neighbour embedding [2], abbreviated to t-SNE is one of the more widely methods for visualizing high dimensional data in usually two or three dimensions. The technique is based on stochastic neighbour embedding [3] but utilising Student's t-distribution.

t-SNE – similarly to all the following algorithms – consists of two main stages. The first one is constructing a probability distribution over pairs of high dimensional objects such that similar points are assigned higher probabilities while dissimilar objects are assigned lower probability. In the second stage, t-SNE constructs a similar probability distribution over the points in the low-dimensional map, then minimizes the Kullback–Leibler divergence between the two distributions with respect to the positions of points in the map. The original version of t-SNE uses Euclidean distance as a metric for similarity, however, this can be changed in popular implementations where a different metric is more appropriate.

Uniform Manifold Approximation and Projection [4], or UMAP for short is a dimension reduction algorithm developed by Leland McInnes, John Healy and James Melville in 2018. It is constructed from a theoretical framework based in Riemannian geometry and algebraic topology. This technique was developed to overcome the shortcomings of t-SNE. The result is an algorithm that is competitive with t-SNE for visualization quality, arguably preserves more global structure and has superior run time performance. UMAP also has no

DOI 10.3311/MINISY2022-016

restrictions on embedding dimension, making it suitable for general dimension reduction use cases.

UMAP works in terms of *fuzzy simplicial sets*, which are higher-dimensional generalizations of directed graphs, partially ordered sets and categories. Indeed, from a practical computational perspective, UMAP can ultimately be described in terms of, construction of, and operations on, weighted graphs. This puts UMAP in the class of k-neighbour based graph learning algorithms such as Laplacian Eigenmaps [5], Isomap [6], or t-SNE.

Both t-SNE and UMAP are considered near-sighted algorithms, meaning they preserve only local structure as opposed to global structure. As PCA can only capture linear dependence relationships, and due to the destructive nature of discarding higher dimensions, it only preserves the highest-level global structure. TriMAP [7] is a graph-based nonlinear dimension reduction algorithm that aims to improve global structure preservation by using triplets of data points instead of pairs.

The algorithm defines triplets (i, j, k) such that $d(\mathbf{x}_i, \mathbf{x}_j) < d(\mathbf{x}_i, \mathbf{x}_k)$. A subset of all triplets $\mathcal{T} \coloneqq \{(i, j, k)\}$ is used to approximate the structure of the dataset.

Pairwise Controlled Manifold Approximation Projection, or PaCMAP [8] for short is the latest dimension reduction algorithm out of the examined ones. It was designed in 2020 by studying the previous three algorithms, uncovering a common interpretation of all of them being graph-based algorithms with nodes being observations and edges constituting a similarity metric between these observations. With this insight, the writers define a set of principles of a good dimension reduction algorithm. They investigate the choice of loss function, the initialization and the algorithm's robustness to initialization. They also investigate the optimization of lower dimensional representation to be able to capture both local and global structure.

The outcome of this investigation is the algorithm called PaCMAP, which satisfies all the principles set by the authors. With the definition of near-pairs, mid-near pairs and further-pairs, the clustering optimization can be affected in such a way as to focus more on either global or local structure.

III. Methodology and Results

A. Outline of Methodology

The examination of the algorithms consisted of five major steps, the first of which was finding already existing implementations of these methods and comparing their computational and usability features. The differences in these features have a major effect on the quality of embedding that can be achieved due to either time constraints or the lack of reproducible and insightful runs.

In the second stage, we ran each algorithm with default parameters to see which implementation offers the best results with relatively small initial effort. This is also good for an indication of what to expect with optimal parameters. This stage resulted in meaningful data of runtimes also.

The third stage was optimizing the hyperparameters of each algorithm. We have selected the most meaningful parameters and performed a sweep from low values to high values to see their effects.

The fourth stage was a linear interpolation test, or LERP for short. By sampling 100 points on a line and running the algorithms with those points in the dataset, a fuller picture can be seen on the mapping of the entire space.

Finally, we investigated the invertibility of each implementation. This is important as targeted searching is more easily executed with sampling points in lower dimension and transforming the sampled points into higher dimension. From the higher-dimensional representation, their respective molecules can be extracted.

B. Dataset

The dataset used in this paper were the 1.6 million molecules in SMILES string format [9] from the ExCape-DB [10] and the Coconut [11] databases along with their molecular properties. The variational autoencoder [12] model from [13] generated the 64 dimensional latent embeddings of the molecules.

The dataset had more features than the SMILES and latent representation of the molecules. Certain chemical descriptors, such as *quantitative estimate of drug-likeness* (QED [14], the distance from a set of known drugs), *synthetic accessibility score* (SaS [15]) among many others. Overall, seven such chemical descriptors were included in the database along with one feature to indicate the database's origin. The last feature was an indicator of whether the data point was in the test set or not during the training of the model. These features – except for the last two – were all utilised by the property predictor of the autoencoder, so the latent space should be relatively smooth over these values.

C. Results

First, we ran each implementation to see how well they perform "out of the box". We also measured the runtime of each implementation when embedding the entire dataset. This is shown in Table I.

It is clear from the data that the choice of algorithm and implementation has a major impact on the runtime, as some implementations offer multithreaded or even GPU features. This however is not the single indicator of the usability of an algorithm, as the quality of the embedding is far more important in assessing the validity.

As can be seen in Figure 1, each algorithm has a unique output space topology. The original dataset was smoothed by regularization over chemical structure, hence every algorithm (with the exception of sklearn t-SNE, nevertheless we include it for completeness) placed the points in an orderly manner. The exact topologies were not equivalent in quality however. During hyperparameter optimization, the most important parameters of each algorithm were examined.

PCA resulted in the smoothest image. This is a problem however, since the quantitative estimate of drug-likeness [14]

Algorithm (iterations)	Runtime	GPU accelerated
PCA	19.45 s	No
t-SNE (sklearn, 150)	90 042 s	No
t-SNE (tsnecuda, 5000)	344.29 s	Yes
UMAP (1000 epochs)	5 866.55 s	Available
TriMAP (2000)	26 756.64 s	No
PaCMAP (2000)	65 283.52 s	No

TABLE I
RUNTIMES OF EACH ALGORITHM TESTED ON THE ENTIRE DATASET (WITHOUT DUPLICATES, 1.6 MILLION MOLECULES) GIVEN IN SECONDS. THE NUMBER OF ITERATIONS AND EPOCHS ARE INDICATED WHERE APPLICABLE. PCA, THE ONLY NON-ITERATIVE METHOD IS THE FASTEST, WHILE THE OTHERS TAKE SIGNIFICANTLY MORE TIME. THE POWER OF GPU USAGE IS CLEARLY DEMONSTRATED BY THE TWO T-SNE IMPLEMENTATIONS.

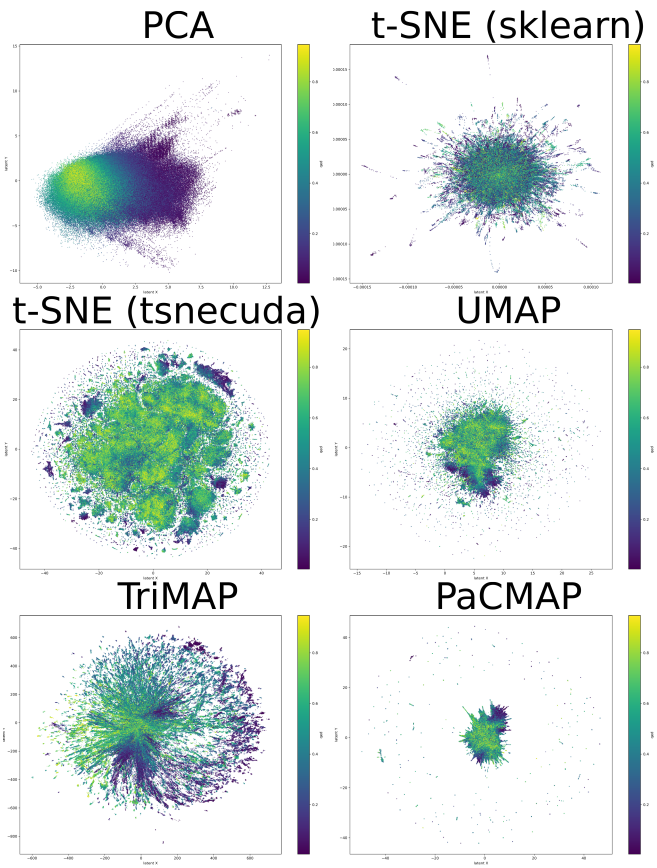


Fig. 1. Results of PCA, t-SNE (sklearn and tsnecuda), UMAP, TriMAP, and PaCMAP algorithms on the dataset with default parameters, coloured by the quantitative estimate of drug-likeness of each data point. It can be clearly seen that each algorithm places points very differently.

is not linearly correlated with the chemical structure of molecules, which makes this embedding unfit for targeted searching. PCA did not have any hyperparameters, thus further investigation was not necessary.

t-SNE performed much better in terms of local structure preservation, as clusters have a smooth gradient over QED value. The greatest weakness of t-SNE is the lack of global structure preservation, leading to similar molecules being placed in different, distant clusters.

TriMAP creates objects resembling magnetic field lines. This is very interesting from a structural view. It should be noted, however, that it seems like the individual clusters overlap with each other in a way that no other algorithm resembles. For this reason, targeted search would be very difficult in the space created by TriMAP. This particular problem, as we will see, does not get better with tuning parameters, making TriMAP fundamentally unusable for this use-case.

Both t-SNE and TriMAP changed very little during the sweep on perplexity and n_inliers respectively. Their potential with different parametrization was on par with the default runs. This is ideal for some applications where there is no need for finding an optimal low-dimensional representation, however, in this use-case it is detrimental. While t-SNE performs well enough for usage, TriMAP with its lack of local structure preservation does not make it a good option.

UMAP clusters the molecules into one big cluster (although not as tightly as PaCMAP), with a few little clusters on the side. This is partly due to the low default min_dist parameter, which makes ultra-tightly packed clusters. The space is very much smooth over QED value. The compactness does make targeted searching more difficult as even a small difference in position can skip multiple molecules, leading to less uniform sampled molecules.

UMAP improved reasonably on increased min_dist values. The ultra-compact clusters disappeared. The overall topology was not changed however.

PaCMAP – similarly to UMAP – produces one compact cluster in the middle, with some additional clusters nearby. This behaviour is not ideal for targeted search, since quite different molecules are placed near each other. PaCMAP does offer very much control over its sightedness however, and with different parameters, this can change.

PaCMAP showed the greatest improvement overall when increasing the FP_ratio parameter, as can be seen in Figure 2. The overall topology of the embedding changed drastically as the parametrization focused more on local structure rather than global. For this reason, PaCMAP holds the greatest potential in this use-case.

The linear interpolation test confirmed the previously stated observations. PCA resulted in a straight line, as it was a linear transformation of the data. t-SNE suffered from the points being thrown into multiple little clusters.

UMAP and TriMAP both formed paths between two points. In the case of TriMAP however, the interpolated points were placed far away from all other points.

PaCMAP was the most significant. As Figure 3 shows, with similar endpoints, the new data was all placed in a cluster, while choosing random endpoints resulted in a path that connected two small clusters.

As for invertibility, the only implementation with such feature support was `umap-learn`. We have briefly tested this and found that targeted searching does work on latent spaces created by this particular implementation.

We can only calculate the embedding of a new point after constructing the embedding with PCA, t-SNE and UMAP.

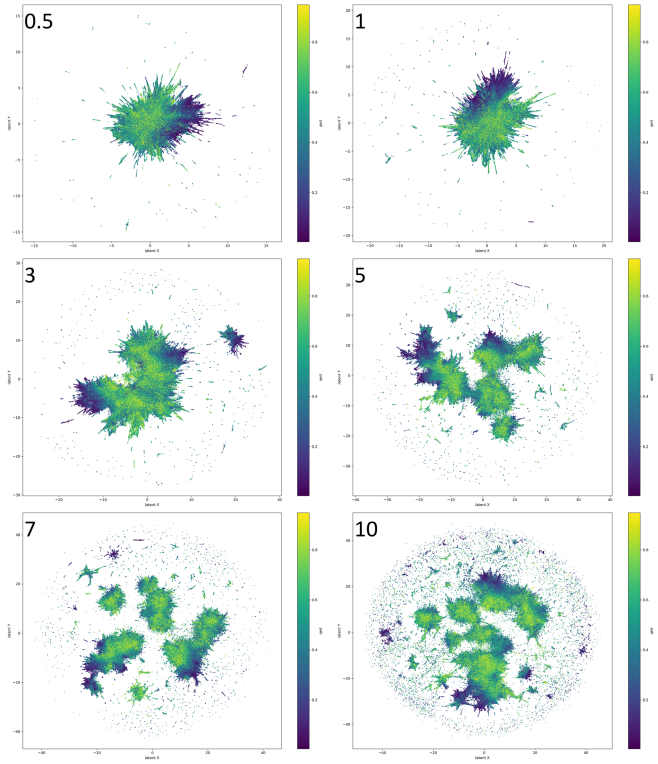


Fig. 2. PaCMAP embeddings using various FP_ratio parameters (0.5, 1, 3, 5, 7 and 10). Higher FP_ratio resulted in more spread out clustering with less similar molecules close together. Tuning this parameter showed the most change out of all the tested non-GD parameters.

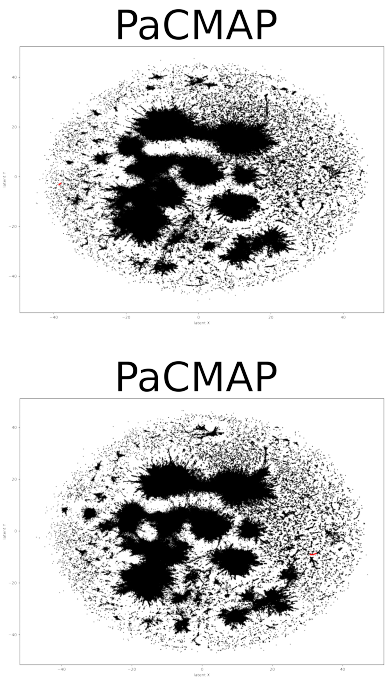


Fig. 3. PaCMAP embedding of the LERP tests. In the first case, the COX-2 inhibitor molecules were all placed on the left of the space in one cluster. The randomly chosen LERP resulted on two clusters being connected on the right. No change to the topology of the whole embedding is observable.

IV. Discussion

The legitimacy of dimension reducing clustering algorithms in *de novo* drug design is of no doubt. We have found that – as many experts already know – PCA is fundamentally unfit for such a task, as it has severe limitations in the correlations it can capture. t-SNE, which to this day is one of, if not the most popular dimension reduction algorithms, suffers from near-sightedness and performance issues both in terms of memory usage and runtime (the latter of which was not an issue because of the GPU implementation). UMAP seems to be among the most useful algorithms in this regard, transforming the chemical space in such a way as to be able to smoothly transition from one molecule to the next. In the tests, PaCMAP showed the most potential as the *de facto* go-to algorithm for pharmaceutical research. Unfortunately, the potential of TriMAP was not explored well enough for any definitive statement. However, educated guesses can be made to suggest that it is suboptimal for allowing targeted searching in its output space.

These findings are invaluable for drug discovery, however, further research is necessary for widespread industrial application.

References

[1] K. Pearson *et al.*, "Liii. on lines and planes of closest fit to systems of points in space," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 559–572, 1901.

[2] L. van der Maaten and G. E. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 2008.

[3] G. E. Hinton and S. Roweis, "Stochastic neighbor embedding," in *Advances in Neural Information Processing Systems* (S. Becker, S. Thrun, and K. Obermayer, eds.), vol. 15, MIT Press, 2003.

[4] L. McInnes *et al.*, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," *ArXiv e-prints*, Feb. 2018.

[5] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[6] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[7] E. Amid and M. K. Warmuth, "TriMap: Large-scale Dimensionality Reduction Using Triplets," *arXiv preprint arXiv:1910.00204*, 2019.

[8] Y. Wang, H. Huang, C. Rudin, and Y. Shaposhnik, "Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization," *Journal of Machine Learning Research*, vol. 22, no. 201, pp. 1–73, 2021.

[9] D. Weininger, "Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules," *Journal of Chemical Information and Computer Sciences*, vol. 28, no. 1, pp. 31–36, 1988.

[10] J. S. et al, "Excape-db: an integrated large scale dataset facilitating big data analysis in chemogenomics," *J Cheminform*, vol. 9, no. 1, p. 17, 2017.

[11] M. S. et al, "Coconut online: Collection of open natural products database," *J Cheminform*, vol. 13, no. 2], 2021.

[12] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2014.

[13] D. Pogány, "Adott célpontra történő gyógyszerhatóanyag generálás hatóanyagmolekulák látens terében," 2020.

[14] R. Bickerton, G. Paolini, J. Besnard, S. Muresan, and A. Hopkins, "Quantifying the chemical beauty of drugs," *Nature chemistry*, vol. 4, pp. 90–8, 02 2012.

[15] P. Ertl and A. Schuffenhauer, "Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions," *Journal of Cheminformatics*, vol. 1, p. 8, Jun 2009.

Using invertible plugins in autoencoders for fast and customizable post-training optimization

Domonkos Pogany
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
pogany.domonkos@gmail.com

Peter Sarkozy
Budapest University of Technology and Economics
Department of Measurement and Information Systems
Budapest, Hungary
psarkozy@mit.bme.hu

***Abstract*—One of the main motivations for modern drug research is the production of new compounds that act as drugs, however developing a new drug is an excessively time and resource intensive process. Deep generative neural networks might provide a solution. With their help, we may be able to search in a continuous latent space to find drug molecules that are not yet known but have suitable chemical and structural properties (e.g. solubility, interaction with a given target protein).**

In this paper, we propose a model which can generate novel drug candidates, that are suitable for a pre-specified objective function of arbitrary properties. The model consists of a generative network and a predictor. The former is an autoencoder which utilizes attention to handle the textual representation of molecules, while the latter uses matrix factorization to predict drug-target interactions (DTI). With a genetic algorithm we can generate novel compounds from the continuous latent space, but if there are changes in the objective function, we may need to train the whole model again. This problem is typical of conditional generative models, to address it, we separated the predictor from the pretrained autoencoder thus forming the plugin. In addition to getting a flexible architecture without any deterioration in the so far achieved results, our model can also be used in a distributed setup by concatenating the plugins. In this way, the objective function can be broken down to smaller subtasks, which can be solved by different plugins without sharing any data.

***Index Terms*—molecule generation, autoencoder, transformer, genetic algorithm, plugin, DTI**

I. Introduction

Finding novel drugs is a difficult challenge, despite all the efforts there are only 10^8 molecules that have been synthesized so far [1] out of the estimated possible 10^{60} drug-like compounds [2]. Developing a new drug is at least a decade-long task usually consisting of three main steps. The first step is the selection of the target proteins, then comes the selection and optimization of the drug compounds, and the last phase is the testing of the drugs. Generative artificial intelligence models may help in reducing the cost and time required for the selection and optimization of drug candidates.

Generating novel drug candidates is an active research field. Most existing methods work with a textual representation of molecules such as the Simplified molecular-input line-entry system (SMILES), which represent molecules as a string. Methods based on a variational autoencoders (VAE) [3] are very common. They can be used to convert the discrete space of the textual representation into a continuous latent space in which the search and optimization tasks can be accomplished more easily. Instead of the textual representation, working on the molecular graphs is also possible [4], this guarantees that the generated molecules will always be syntactically valid. Other methods construct novel and valid molecules step by step with the use of reinforcement learning [5], or by starting from a known compound and optimizing it according to a molecule property using a generative adversarial network (GAN) [6]. To handle the input representation, most models use recurrent layers. Using attention layers instead, to speed up the training process is also becoming a popular choice. Generating compounds with transformers [7] have shown promising results lately [8].

The methods mentioned so far can all generate valid molecules according to some chemical properties, however, they do not in themselves provide a solution for generating a pharmacokinetically active molecule that interacts with the target protein. Measuring all interactions between every compound-target pair is extremely costly, thus we need to use estimations. Neural networks can also be used for this task, for instance we can train a convolutional network to take a molecule and a target as an input and predict the interaction on the output [10]. By combining the generative and the predictive models above, in theory we are able to generate drug candidates for a given target.

Recent research places great emphasis on the development of a selective binding profile. The goal is to generate molecules that bind not only to one but to several target proteins. Moreover, there are targets we want to avoid binding to because of the possible side effects. Knowing the targets, we can now utilize the methods mentioned above to generate drug candidates. For instance, a VAE based method was presented to find compounds to treat psychiatric disorders by generating novel targets to the indicated genes [11].

Using these models in practice, however, introduces new problems. When modifying the objective function, such as introducing a new target, we might need to train the entire model again which could take weeks. Consequently, we need to utilize pre-trained models. More than that, most of the times, we do not have sufficient training data available. Usually compounds and even targets under research are kept highly

DOI 10.3311/MINISY2022-017

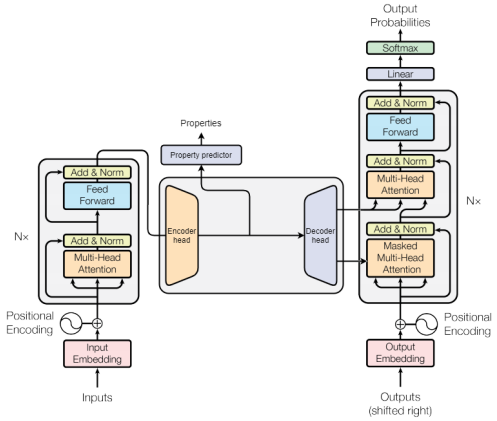


Fig. 1. Transformer based autoencoder [7].

confidential by pharma entities. The goal is to develop methods that allow partners to benefit from each other's knowledge without having to share their data and results.

In this paper we propose a model which can provide a solution to the above-mentioned problems. Besides being able to generate novel molecules for a multi-target objective function, training the model to a new, and previously not included target is highly efficient, and it can even be used in a distributed environment without sharing any valuable data.

II. Implemented methods

A. Transformer-based autoencoder for novel compounds

We chose to work with the SMILES representation of molecules. We used a transformer network to handle the textual representation, and to be able to generate new compounds, we converted it into an adversarial autoencoder (AAE) [12] which uses a Gaussian prior.

First, the input molecules are encoded into a continuous latent space, to achieve this we put an extra attention layer in top of the original transformer blocks. From there chemical properties are predicted using fully connected layers. By propagating the error of the predictor back through the encoder, molecules with similar properties are mapped close together in the latent space. The decoder consists of transformer decoder blocks, each of them receives information from the latent space through fully connected layers. Two different representations are decoded from the latent space, one going into the cross attention layer and another into the self attention layer. The latter is concatenated into the decoder-side representations of the characters from the previous layer, thus forming a pseudo attention layer [13]. Fig. 1 shows the architecture of the model without the discriminator network.

From now, the problem of generating molecules to a given target function can be solved by searching in the latent space. To perform the search, we implement a genetic algorithm. The individuals are the molecules, their genes are their latent representations, and the fitness function is the objective function which we want to maximize. With this choice we can easily implement mutation as additive noise, and crossover as interpolation in the latent space.

B. Matrix factorization plugin for generating to a given target

To be able to generate drug candidates to a given set of proteins, first we need to somehow obtain and process DTI data. We chose to work with the ExCAPE-DB dataset [14] as it contains $\sim$600K molecules and $\sim$1300 targets. To utilize this extra information, we use an interaction predictor the same way as the property predictor. It solves the problem of the prediction with matrix factorization by taking the product of the latent molecule representation and the learned target embedding. As DTI data is often imbalanced with respect to active vs. inactive molecules, we weighted the binary cross-entropy loss with the imbalance ratio.

With the extensions discussed earlier we can generate molecules to maximize an objective function which consists of the predicted binding affinity to multiple target proteins. However, the end-to-end training of the model has a lot of drawbacks in practice. As already mentioned, it is inefficient when generating to a new target protein, and the sparsity of the available interaction data can also be a problem. To alleviate these problems, we pretrain our model. There are several pre-training techniques, and we have further investigated and developed a method created for conditional autoencoders [15]. The main idea is to pretrain the autoencoder and freeze all the weights, then attach a plugin into the bottleneck, which is a smaller autoencoder using the latent representation as input and has its own, inner latent space. This way the reconstruction and the conditional generation tasks can be separated, and when we receive a new condition, we only need to change the plugin. For the reconstruction task, we merged the GuacaMol, ExCAPE-DB and the DrugSpaceX [16] datasets, thereby resulting in 9.2 million training molecules for the model. After pretraining, the ExCAPE-DB dataset can be used to train the plugin and the interaction predictor. The predictor now uses the inner latent space as the input therefore the interaction data is encoded into it. Due to the small number of parameters in the plugin, we can learn a novel target in a relatively short time, moreover we can achieve good generalization even with a small amount of available interaction data. Plugins simplify post-training on any subset of the full training dataset. The modified architecture is shown in Fig. 2, the encoder and decoder of the plugin both consists of fully connected layers with a leaky ReLU between them.

After training the plugin the model was able to predict interactions from the inner latent space but lost the ability to reconstruct molecules from the pretrain dataset (from 95% to 10%). To solve this, we replaced the plugin decoder with the inverse of the plugin encoder. Therefore, the reconstruction power of the pretrained model is preserved. Moreover, training the plugin is greatly accelerated because it is no longer has trainable parameters in the decoder, so the decoder of the plugin and the transformer is not utilized during training. The

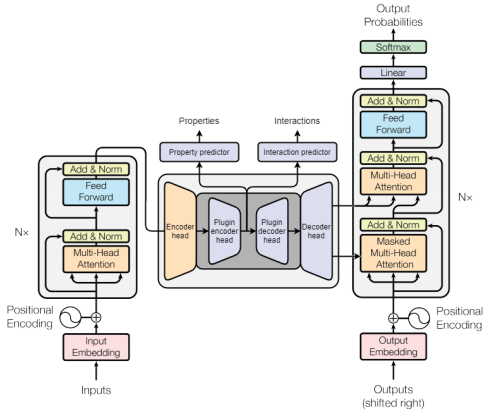


Fig. 2. Pretrained autoencoder extended with a plugin.

invertibility of the plugin carries a few restrictions[1], and the model becomes increasingly sensitive to noise. The hidden representation of the encoder in layer l, $x^{(l)}$ can be calculated as

$$x^{(l)} = g(W^{(l)}x^{(l-1)} + b^{(l)}) \quad \text{with} \quad g(x) = \begin{cases} x, & \text{if } x \geq 0 \\ \frac{x}{2}, & \text{otherwise,} \end{cases} \tag{1}$$

where $b^{(l)}$ is the bias, $W^{(l)}$ is the weight matrix in layer l, and g denotes the activation function. Due to invertibility $W^{(l)}$ must be a square matrix, g must be invertible and its range must not be finite, hence we use a leaky ReLU. This way a layer in the decoder takes the form below

$$x^{(l)} = W^{(l)+}(g^{-1}(x^{(l-1)}) - b^{(l)}), \tag{2}$$

where $W^{(l)+}$ denotes the Moore–Penrose pseudoinverse of the weight matrix[2].

However, large weights in the matrix amplify sensitivity to noise in the plugin, even a small amount of noise added to the inner latent space resulted in high distances in the outer space. We examined the weight matrices and noticed that this can be a result of the high condition numbers ($> 1e4$). The sensitivity to noise can be handled by controlling the condition number of the components in the plugin network. Conventional regularization methods such as L1 and L2 loss were unsuccessful in keeping the condition number low, so we added the condition number directly to the loss function as a regularization parameter, and determined that condition numbers < 20 were still stable with respect to noise. The condition number of the weight matrix upon initialization was also controlled. We also found that using a steeper ReLU slope leads to a more stable training.

[1]The invertibility of the encoder was also the main reason behind choosing AAE, because the reparametrization trick used in a VAE is not invertible.

[2]It is very unlikely that the weight matrix is not invertible, and there are also methods with which we can assure the invertibility of the learned weights, for example learning the QR decomposition of the matrices.

The components of the loss function used for training the autoencoder plugin are the weighted and weight-normalized sum of the following:

- categorical cross-entropy of the reconstruction,
- binary cross-entropy of the DTI classification,
- MSE of the molecule property predictors,
- L2 weight regularization,
- condition number regularization: $max(0, C-20)^2$, where C is the condition number of the weight matrices,
- adversarial loss.

With these modifications the model was indeed able to generate hundreds of novel molecules from the inner latent space that were suitable for a given objective function.

C. Serial plugin for generating in a distributed environment

With this flexible architecture when a new protein is added to the targets only the plugin needs to be retrained. We can go even further by having several plugins to several targets. This way when a new target is required, a new plugin should be trained, the other plugins remain intact, so the time previously invested in them is not wasted. When generating we now have several plugins, each contributes to a couple of predicted properties or interactions. This is equivalent to connecting them serially, as inversion preserves the input. Fig. 3 shows the serial architecture.

To implement the method, besides connecting the plugins the optimization process based on the genetic algorithm needs to be modified. The fitness function can be constructed from the output of the plugins. The only difference with the one plugin case is that there are now more latent spaces, so we need to come up with a new crossover method. One solution is for each plugin to generate some individuals to form the new generation, but it can also work if we generate the next generation with a different plugin in each round. Selection and mutation can be performed as usual in the outer latent space.

The separability of the architecture and the versatility of the genetic algorithm allows the model to be used in a distributed environment. The obvious solution is to give every partner a plugin to work with, this way they only need to agree on the

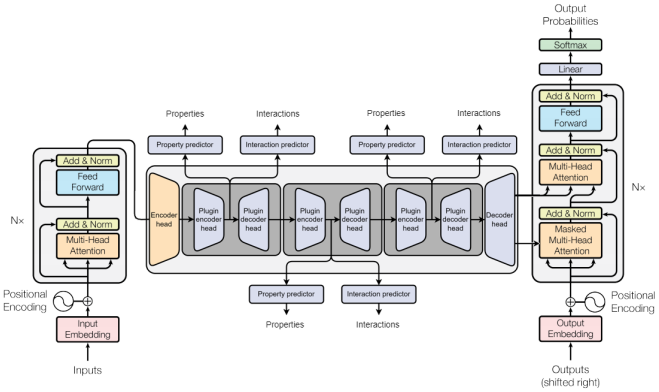


Fig. 3. Autoencoder with several plugins connected in series. Invertibility permits the usage of any number of plugins in serial connection.

pretraining data set. During generation they need to share their own calculated part of the fitness function and the molecules generated for the next generation. The sensitive training data, the known interactions, the parameters and architecture of the model, and even the methods using for prediction can remain hidden, mitigating the most common federated attacks [18]. More than that, the partners do not even need to share their selected target proteins if they are using a more abstract fitness function, e.g. avoiding a given side effect instead of not to bind to a given set of targets.

III. Results

We trained the model on the GuacaMol dataset [9] which contains roughly 1.6 million drug-like molecules, to evaluate the performance of the autoencoder model before attaching plugins. After the optimization of the hyperparameters, we managed to reach a remarkably high, 97.2% reconstruction on the test molecules and obtained similar benchmark results as the other models reported by the GuacaMol benchmark [9]. Using a sampling method of a random noise vector with $\mu = 0.0$, $\sigma = 0.25$ around test data, we achieved a validity of 96.11%, a uniqueness of 99.97%, a novelty of 97.18%, and the KL and FCD scores were 0.9688 and 0.8505 respectively. To test the plugins, we generated drug candidates to bind to specific targets. We separated the problem into two subtasks, to generate drug-like and synthesizable molecules and to generate molecules which can bind to a set of related protein targets, that often have strong cross-binding for most known drug compounds. First, we had pretrained the model on our merged dataset, after that, two plugins were trained. The one, which was trained on the GuacaMol dataset was responsible for the first task. To achieve this, we used a property predictor to predict the quantitative estimate of drug-likeness (QED) and synthetic accessibility score (SAS) of the molecules. A compound with higher QED is more drug-like, while lower SAS means it can be more easily synthesized. The other plugin was trained on the ExCAPE-DB dataset with an interaction predictor to predict the binding to a set of 3 related genes. We used a weighted sum of the desired molecule properties and target binding affinities as an objective function.

We were able to generate hundreds of novel and valid compounds which were not presented in any of the three datasets. We tested promising candidates with high fitness scores and made interaction predictions with the Swiss Target Prediction [17] software, and the molecules showed high binding affinity to the selected targets.

While the average target prediction accuracy did not improve significantly from the model trained without the plugin architecture, but the additional benefit of using an arbitrary number of plugins trained for different tasks allows rapid customization of desired target profiles, compared to retraining the entire network.

IV. Conclusion and future work

We have seen the challenges inherent in drug discovery and that the combination of generative and predictive models to reach a selective binding profile plays an increasingly important role in this field of research.

In our paper we introduced a method which can be used to generate suitable candidates for an arbitrary objective function. By separating the task of reconstruction and conditioned generation we end up with a model capable to efficiently handle newly arriving targets. Finally, we shoved how it can be used in a distributed environment without sharing any confidential data.

For further improving the model we want to try the universal transformer architecture [19] and the relative position embedding [20].

References

[1] S. Kim et al., “PubChem substance and compound databases”, Nucleic acids research, vol 44, no D1, bll D1202–D1213, 2016.

[2] P. G. Polishchuk, T. I. Madzhidov, en A. Varnek, “Estimation of the size of drug-like chemical space based on GDB-17 data”, Journal of computer-aided molecular design, vol 27, no 8, bll 675–679, 2013.

[3] R. Gómez-Bombarelli et al., “Automatic chemical design using a data-driven continuous representation of molecules”, ACS central science, vol 4, no 2, bll 268–276, 2018.

[4] W. Jin, R. Barzilay, en T. Jaakkola, “Junction tree variational autoencoder for molecular graph generation”, in International conference on machine learning, 2018, bll 2323–2332.

[5] Z. Zhou, S. Kearnes, L. Li, R. N. Zare, en P. Riley, “Optimization of molecules via deep reinforcement learning”, Scientific reports, vol 9, no 1, bll 1–10, 2019.

[6] Ł. Maziarka, A. Pocha, J. Kaczmarczyk, K. Rataj, T. Danel, en M. Warchoł, “Mol-CycleGAN: a generative model for molecular optimization”, Journal of Cheminformatics, vol 12, no 1, bll 1–18, 2020.

[7] A. Vaswani et al., “Attention is all you need”, in Advances in neural information processing systems, 2017, bll 5998–6008.

[8] R. Irwin, S. Dimitriadis, J. He, en E. J. Bjerrum, “Chemformer: A Pre-Trained Transformer for Computational Chemistry”, Machine Learning: Science and Technology, 2021.

[9] N. Brown, M. Fiscato, M. H. S. Segler, en A. C. Vaucher, “GuacaMol: benchmarking models for de novo molecular design”, Journal of chemical information and modeling, vol 59, no 3, bll 1096–1108, 2019.

[10] H. Öztürk, A. Özgür, en E. Ozkirimli, “DeepDTA: deep drug–target binding affinity prediction”, Bioinformatics, vol 34, no 17, bll i821–i829, 2018.

[11] X. Tan et al., “Automated design and optimization of multitarget schizophrenia drug candidates by deep learning”, European Journal of Medicinal Chemistry, vol 204, bl 112572, 2020.

[12] A. Makhzani, J. Shlens, N. Jaitly, I. Goodfellow, en B. Frey, “Adversarial autoencoders”, arXiv preprint arXiv:1511. 05644, 2015.

[13] L. Fang, T. Zeng, C. Liu, L. Bo, W. Dong, en C. Chen, “Transformer-based Conditional Variational Autoencoder for Controllable Story Generation”, arXiv preprint arXiv:2101. 00828, 2021.

[14] J. Sun et al., “ExCAPE-DB: an integrated large scale dataset facilitating Big Data analysis in chemogenomics”, Journal of cheminformatics, vol 9, no 1, bll 1–9, 2017.

[15] Y. Duan, C. Xu, J. Pei, J. Han, en C. Li, “Pre-train and plug-in: Flexible conditional text generation with variational auto-encoders”, arXiv preprint arXiv:1911. 03882, 2019.

[16] T. Yang et al., “DrugSpaceX: a large screenable and synthetically tractable database extending drug space”, Nucleic acids research, vol 49, no D1, bll D1170–D1178, 2021.

[17] A. Daina, O. Michielin, en V. Zoete, “SwissTargetPrediction: updated data and new features for efficient prediction of protein targets of small molecules”, Nucleic acids research, vol 47, no W1, bll W357–W364, 2019.

[18] L. Lyu, H. Yu, en Q. Yang, “Threats to Federated Learning: A Survey”, arXiv [cs.CR]. 2020.

[19] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, en Ł. Kaiser, “Universal transformers”, arXiv preprint arXiv:1807. 03819, 2018.

[20] P. Shaw, J. Uszkoreit, en A. Vaswani, “Self-attention with relative position representations”, arXiv preprint arXiv:1803. 02155, 2018.

ZKP-Based Audit for Blockchain Systems Managing Central Bank Digital Currency

Bertalan Zoltán Péter, Imre Kocsis
Department of Measurement and Information Systems
Budapest University of Technology and Economics
Budapest, Hungary
bpeter@edu.bme.hu, ikocsis@mit.bme.hu

***Abstract*—Central Bank Digital Currency (CBDC) systems are being developed around the world and production solutions can be expected in the near future. Should a central bank allow handling of CBDC on a ledger that is not under its supervision (via platform bridging), it may wish to specify certain conformance requirements regarding the transactions. We propose a novel audit scheme based on Zero-Knowledge Proofs, which allows the operator of the bridged ledger to prove its compliance to such requirements, without revealing details about the transactions (such as the exact participants, the direction of the transfer, or the transferred value). This scheme aims to resolve the conflict between banks having to audit how CBDC is used on the bridged blockchain and consortia trying to keep sensitive data private.**

***Index Terms*—blockchain, audit, central bank digital currency, zero-knowledge proof**

I. Introduction

Central Banks (CBs) around the world are actively researching the possibilities of implementing their own Central Bank Digital Currency (CBDC), with some institutions already testing prototypes and pilots [1]. Considering the popularity and success of cryptocurrencies and blockchains equipped with various smart contracts, we can safely assume that once a production-grade CBDC emerges, there is soon going to be a real demand to make it usable on private as well as public blockchain networks. For example, a consortium might maintain their own blockchain network, where they wish to use CBDC as currency. A large motivating factor is that the CBDC system most likely will not allow the installation of arbitrary smart contracts (or universal programmability of any form), a feature widely used by cryptocurrencies today.

There is already a solution known in the cryptocurrency world for the integration of platforms that do not support smart contracts and those that do support them, in the form of platform *bridging*. Bridging is essentially the process of allowing some units (in our case, CBDC) to be locked on one ledger, while so-called *shadow* or *wrapped* units of equivalent value are created on the other ledger (the *side-chain*). The shadow can later be converted back into the original instrument on the first ledger, possibly by a new owner.

However, such bridged CBDC would still be digital money which has to adhere to certain Know Your Customer / Anti Money Laundering (KYC/AML) requirements established by law. A simple example is requiring CBDC (or its shadow) to only be owned by persons or organizations who are allowed (ie whitelisted) at all times. In other words, digital cash cannot get into the hands of forbidden parties. To prove that these requirements are satisfied, the transactions done with the bridged CBDC will most likely be subject to audits. Unfortunately, the straightforward option of the auditors simply reading the ledger contents is not viable since the information in the ledger may be highly confidential. The bank is not interested in all of the data in the ledger, merely in compliance with the requirements.

In this paper, we present a novel audit scheme for the bridged CBDC scenario, which makes it possible for the CB to verify conformance to arbitrary requirements in such a way that potentially confidential information stored on the blockchain is not revealed. Our specific contributions are the following:

- We establish and formalize a minimal blockchain model for our purposes.
- We define five elementary audit criteria, which must be met by the transactions on the blockchain.
- We define a complete audit scheme, which allows a bridged blockchain to prove conformance to these criteria.

The rest of this paper is organized as follows. We further elaborate on CBDC bridging in the subsection below, followed by comparing our approach to other recent work in section II. Section III introduces our simplified blockchain model. Building on this formalization, we present an audit model and protocol in section IV, which we have implemented as a prototype. Finally, section V summarizes our results and plans for future work.

CBDC Bridging

CBDC is a form of digital fiat money issued by CBs, just like physical banknotes that have existed for hundreds of years. It is similar to other forms of digital money in the sense that it requires maintaining a *ledger* of transactions or at least a registry of current account balances [2]. How this ledger is accessed, structured, and maintained is a design choice.

A blockchain is essentially a data structure consisting of a cryptographically linked list of *blocks* that contain *records.*

The research was supported within the framework of the Cooperation Agreement between the National Bank of Hungary and BME.

DOI 10.3311/MINISY2022-018

While cryptocurrencies traditionally store their ledger in blockchains, CBDC implementations may call for a centralized ledger solution, not a distributed one. Nevertheless, blockchain systems are still relevant in the context of CBDC when we consider the possibility of 'transferring' CBDC to an external blockchain network (sometimes called a *side-chain*) where it may be used similarly to cryptocurrencies.

The usual definition of CBDC does not imply universal programmability by users. In the context of Distributed Ledger Technology (DLT) and blockchains, this means that CBDC implementations will likely not support arbitrary smart contracts. On the other hand, smart contracts are prevalent in today's cryptocurrencies and are drivers of innovation in several areas. However, the fact that they cannot handle digital cash (as opposed to cryptocurrencies or stablecoins) remains largely unaddressed. One way to integrate smart contracts with CBDC is to adopt the existing *bridging* technology for integrating different blockchain platforms. Figure 1 offers a simple architectural overview of such a bridged CBDC scenario.

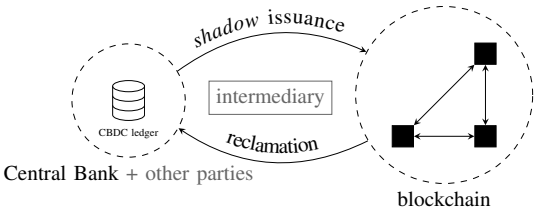


Fig. 1: CBDC bridging

The question is: in this setup, can the CB enforce requirements and conduct periodic or on-demand audits on the consortial blockchain *without* gaining access to sensitive information in the process? In this paper, we focus on an audit model and protocol based on a relatively new family of cryptographic algorithms, Zero-Knowledge Proofs (ZKPs), which essentially allow the bank to verify – or, from another perspective, the consortium to *prove* – that the requirements are satisfied without seeing into any of the transactions, or even their number.

II. Related Work

zkrpChain [3] and *zkLedger* [4] are somewhat similar works, but neither target nor solve the exact same problem. As its name suggests, the former focuses on *range proofs*, which are not readily applicable to the verification of criteria such as being on a whitelist (set membership). The implementation mainly uses smart contracts. On the other hand, our proposal is universal because it allows arbitrary computations to verify the requirements. Thus, anything be expressed as a program (source code) can be verified.

zkLedger requires the auditees to actively participate in the audits and maintain a synchronized *commitment cache*. In the audit scheme outlined in this paper, however, the participating organizations of the side-chain need not actively take part in audits; a single blockchain node suffices.

Additionally, neither of these solutions offers an easily programmable interface to define the audit criteria. We have implemented a prototype in a ZKP system that allows the expression of the requirements as a simple, procedural program (rather than arithmetic circuits, for example).

III. Blockchain and Requirement Model

To rigorously define an audit protocol, we must first establish what we mean by the blockchain to which CBDC is bridged.

A. Blockchain

The blockchain is an infinitely growing sequence of blocks, where each block consists of a block header (denoted by B) and a block body (denoted by T). The header contains data that makes this construction a blockchain, and the body is simply a sequence of transactions organized as a *Merkle tree* [5] which belong to the block. Figure 2 is a visual representation of a segment of the block sequence. We index block headers and bodies as B_i and T_i respectively, denoting the header and body of the i-th block (ie the block at height i).

Thus, the entire blockchain $\mathfrak{B}$ at a given height (ie block count) h can be expressed as a sequence of ordered pairs of B_i and T_i: $\mathfrak{B} = ((B_0, T_0), \ldots, (B_h, T_h))$.

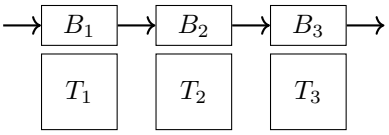
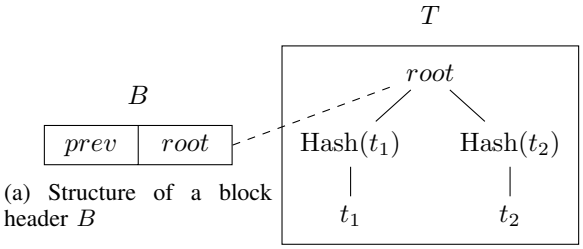


Fig. 2: The block sequence of the blockchain model

1) Blocks: Each block consists of a header and a body. The header is an ordered pair of the *hash* of the previous block's header ($prev$) and the root of the Merkle tree that contains all the transactions in the block ($root$). The body is the sequence of transactions stored in a *Merkle tree*, whose top hash is $root$. Figure 3 illustrates the individual block headers and bodies. The *genesis* block's $prev$ value is nil. We denote the sequence of all blocks in the blockchain $\mathfrak{B}$ by $\mathbb{B}_{\mathfrak{B}}$.



(a) Structure of a block header B

(b) Structure of a block body T

Fig. 3: Visualization of block headers and bodies in the blockchain model

2) Transactions: Transactions (denoted by t) are ordered triples formed by their *source* s, their *receiver* r and the *amount* a of funds transferred.

To further simplify the model, let us assume that each block contains exactly two transactions:

$$\forall (B = (root, prev), T = (t_1, t_2)) \in \mathfrak{B} \\ : root = \mathrm{Hash}\left(\mathrm{Hash}(t_1) + \!\!+ \mathrm{Hash}(t_2)\right)$$

'Hash' denotes an arbitrary collision-resistant hash function and $+\!\!+$ is the concatenation operator (applied to binary values).

The sequence of transactions in a blockchain $\mathfrak{B}$ – denoted by $\mathbb{T}_{\mathfrak{B}}$ – is understood as the sequence of all transactions in all blocks: $\mathbb{T}_{\mathfrak{B}} = \{t \in T : (B, T) \in \mathfrak{B}\}$.

We also define $\mathcal{S}(t)$, $\mathcal{R}(t)$, and $\mathcal{A}(t)$ to denote the sender, receiver, and amount of a transaction t respectively.

3) Accounts: Accounts represent the senders and receivers of transactions. In our model, they are simply identified by integer values: $\mathbb{A} \neq \varnothing \subseteq \mathbb{N}$. Account 0 is the *genesis account*, which always exists. We assume that the set of accounts is constant for the lifetime of the blockchain.

Every account has its *balance*, which is a non-negative integer. When the blockchain is created, the genesis account's balance becomes the *total balance* in the blockchain. In other words, there is a fixed amount of units ('money') in the system at all times.

We denote the set of all accounts in a blockchain $\mathfrak{B}$ by $\mathbb{A}_{\mathfrak{B}}$.

4) Genesis: Genesis refers to the creation of a blockchain. The following two parameters are involved when creating a blockchain $\mathfrak{B}$: the total balance $\$_{\mathfrak{B}}$, and the final set of accounts $\mathbb{A}_{\mathfrak{B}}$.

B. Requirements

We have collected five basic requirements to which blockchain state is expected to conform, verified during an audit protocol. We want to ensure that for all transactions in any given block

(1) the sender has sufficient balance to spend;
(2) the receiver is allowed to receive funds (ie is whitelisted);
(3) the balance of the receiver after the transaction equals their balance before the transaction *plus* the transferred funds;
(4) the balance of the sender after the transaction equals their balance before the transaction *minus* the transferred funds;
(5) the hash of the block header (found in the next block's header) is indeed the hash of the block's header concatenated with the root of the Merkle tree that contains the transactions in the block.

Based on our formalized blockchain model, we can express these requirements succinctly, as seen in Table I.

t^i_j denotes the jth transaction in the ith block B_i. $\mathcal{P}(t)$ is the transaction immediately before t:

$\mathcal{P}(t^i_j) = \begin{cases} t^i_{j-1} & : j > 1 \\ t^{i-1}_N & : \text{otherwise} \end{cases}$, where N is the (constant) number of transactions in each block (defined to be 2 for simplicity). $\mathcal{B} : \mathbb{A} \times \mathbb{T} \to \mathbb{N}$ with $\mathcal{B}(u, t^i_j) =$ the balance of account u *after* transaction t^i_j is a function defined to simplify expressing account balances.

req.	formalization
(1)	$\forall t^i_j \in \mathbb{T} : \mathcal{B}\left(s, \mathcal{P}(t^i_j)\right) \geq \mathcal{A}(t^i_j)$
(2)	$\forall t^i_j \in \mathbb{T} : \mathcal{S}(t^i_j) \in WhiteList$
(3)	$\forall t^i_j \in \mathbb{T} : \mathcal{B}(r, t^i_j) = \mathcal{B}\left(r, \mathcal{P}(t^i_j)\right) + \mathcal{A}(t^i_j)$
(4)	$\forall t^i_j \in \mathbb{T} : \mathcal{B}(s, t^i_j) = \mathcal{B}\left(s, \mathcal{P}(t^i_j)\right) - \mathcal{A}(t^i_j)$
(5)	$\forall B_i = (prev_i, root_i), B_{i-1} = (prev_{i-1}, root_{i-1}) \in \mathbb{B}$ $: prev_i = \mathrm{Hash}(B_{i-1}) = \mathrm{Hash}(prev_{i-1} + \!\!+ root_{i-1}) \quad (i > 0)$

TABLE I: Formalized requirements

$$\mathcal{B}(u, t^i_j) = \sum_{n=0}^{i-1} \left(\sum_{\theta \in T_n : \mathcal{R}(\theta) = u} \mathcal{A}(\theta) - \sum_{\theta \in T_n : \mathcal{S}(\theta) = u} \mathcal{A}(\theta) \right) \\ + \sum_{k=1}^{j} [\mathcal{R}(t^i_k) = u]\, \mathcal{A}(t^i_k) - \sum_{k=1}^{j} [\mathcal{S}(t^i_k) = u]\, \mathcal{A}(t^i_k)$$

where $[P]$ is an *Iverson-bracket* expression, ie $[P]$ is 1 if P is true and 0 otherwise. $WhiteList \subseteq \mathbb{A}$ is a set of allowed transaction senders, which may change over time.

IV. Audit Scheme

During an audit, the CB must be able to verify conformity to the requirements without directly accessing ledger contents. As we established earlier, the latter may not be compatible with the confidentiality requirements of the consortium using the blockchain. Our audit scheme relies on ZKPs to allow the consortium to prove that the ledger state is valid in *zero-knowledge*. In other words, the bank (the auditor) learns nothing more in the process other than that the requirements are satisfied.

ZKPs are cryptographic constructions that allow a Prover to prove the truth of a statement to a Verifier in such a way that no information is revealed in the process other than whether the statement is true. They are relatively new in mathematics and cryptography but already have several applications in cryptocurrencies where privacy is paramount. For example, *Zerocoin* [6] and its successor, *Zerocash* [7], heavily rely on ZKPs.

As shown by [8], it is possible to generate a Non-Interactive ZKP for arbitrary *computations* done on a von Neumann architecture. This essentially means that a program or algorithm itself can be the subject of a Zero-Knowledge Proof. For our purposes, the audit protocol can be expressed as a computation that verifies our requirements in zero-knowledge.

Several ZKP systems and software implementations exist. In our work, we used *Zilch* [9], mainly because it allowed rapid prototyping thanks to its Java-like procedural language in which computations can be expressed. *Zilch* internally relies on Zero-Knowledge Succinct Transparent ARguments of Knowledge (zk-STARKs).

Our audit protocol design is *interactive*, meaning that the CB and the blockchain engage in an online, synchronous exchange of messages, during which the satisfaction of the requirements by the ledger state is proven. The interactive audit flow is visualized in Figure 4.

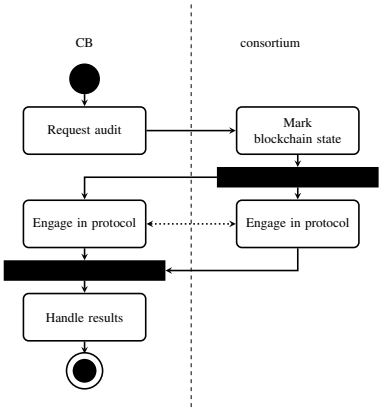


Fig. 4: Interactive audit

A crucial problem to solve is how to ensure that the state which forms the subject of the audit is the actual ledger state, and not data fabricated by a malicious consortium. A simple solution would be for the CB to maintain their own so-called *lightweight* or *header* node on the blockchain, which does not have access to block bodies (ie transactions), only headers. Thus, no information is leaked by the block headers themselves. In such a setup, an export of the marked state would be committed to the blockchain state in the sense that it contains the hashes of all blocks (except perhaps the last one), which are also recomputable for each block from the transactions.

The rest of this section describes an audit process in our model. Let us start with an arbitrary state of the side-chain and an incoming audit request from the CB. The current state is marked: the audit is performed on the data that existed at the time of the audit request. In the meantime, the blockchain can still be operational; blocks can be appended, smart contracts can be executed. From this point, the required steps are the following:

1) If required, we convert the blockchain state into a format that can be fed into the audit algorithm. In this process, we trim all unnecessary information and essentially generate a *view* of blockchain, which is just a sequence of transactions.
2) We open a communications channel between the CB and a node on the blockchain to convey audit information.
3) The two parties engage in an Interactive Zero-Knowledge Protocol, during which the verification algorithm is executed as a computation, and successful termination signifies a successful audit. This algorithm must be previously agreed upon by both parties and is itself public. The concrete arguments of the algorithm are only known by the blockchain node.
4) Whether successful or not, the CB takes note of the event.

It is up to the CB to decide how the results of a non-compliant audit are handled: they may require a more in-depth audit to give the consortium a chance to come clean about the situation at the cost of exposing their private data, or they may simply record the violation.

More information, including the verification protocol as pseudocode and the *Zilch* prototype can be found in [10].

V. Conclusion and Further Work

In our work, we have presented a complete audit model and protocol for a bridged CBDC scenario using ZKPs. We have created a prototype implementation using *Zilch*, which showed promising results during ad-hoc testing with hand-crafted data. After improving on the less refined parts of our implementation, we would like to put it to the test by evaluating its performance on data similar in volume to what is expected on a real consortial network.

An exciting challenge is how bridged CBDC that is processed by smart contracts can be handled. Tackling this problem would take us one step closer to real-life applications. We also plan to consider the transactions' cryptographic signatures, verifying them during audits.

Looking further, the audit protocol outlined in this paper is not specialized to CBDC systems whatsoever. The prospects of applying the methodology in the broader area of cross-organizational integrations are certainly worth looking into.

References

[1] "CBDC tracker," https://cbdctracker.org, accessed: 2022-01-11.
[2] C. Boar, H. Holden, and A. Wadsworth, *Impending arrival - a sequel to the survey on central bank digital currency*, ser. BIS Papers. Bank for International Settlements, 04 2020, no. 107. [Online]. Available: https://ideas.repec.org/b/bis/bisbps/107.html
[3] S. Xu, X. Cai, Y. Zhao, Z. Ren, L. Wu, H. Zhang, L. Du, and Y. Tong, "zkrpchain: Privacy-preserving data auditing for consortium blockchains based on zero-knowledge range proofs," in *2020 IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2020, pp. 656–663.
[4] N. Narula, W. Vasquez, and M. Virza, "zkledger: Privacy-preserving auditing for distributed ledgers," in *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. Renton, WA: USENIX Association, 04 2018, pp. 65–80. [Online]. Available: https://www.usenix.org/conference/nsdi18/presentation/narula
[5] R. C. Merkle, "A digital signature based on a conventional encryption function," in *Advances in Cryptology — CRYPTO '87*, C. Pomerance, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1988, pp. 369–378.
[6] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous distributed e-cash from bitcoin," in *2013 IEEE Symposium on Security and Privacy*, 2013, pp. 397–411.
[7] E. Ben-sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from bitcoin," in *2014 IEEE Symposium on Security and Privacy*, 05 2014, pp. 459–474.
[8] E. Ben-Sasson, A. Chiesa, E. Tromer, and M. Virza, "Succinct non-interactive zero knowledge for a von neumann architecture," in *Proceedings of the 23rd USENIX Conference on Security Symposium*, ser. SEC'14. USA: USENIX Association, 2014, pp. 781–796.
[9] D. Mouris and N. G. Tsoutsos, "Zilch: A framework for deploying transparent zero-knowledge proofs," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 3269–3284, 2021.
[10] B. Z. Péter, "ZKP-based audit for blockchain systems managing central bank digital currency," 2021, *Scientific Student Competition* Report. [Online]. Available: https://tdk.bme.hu/VIK/inform/ZKPalapu-audit-digitalis-jegybankpenzt-kezelo