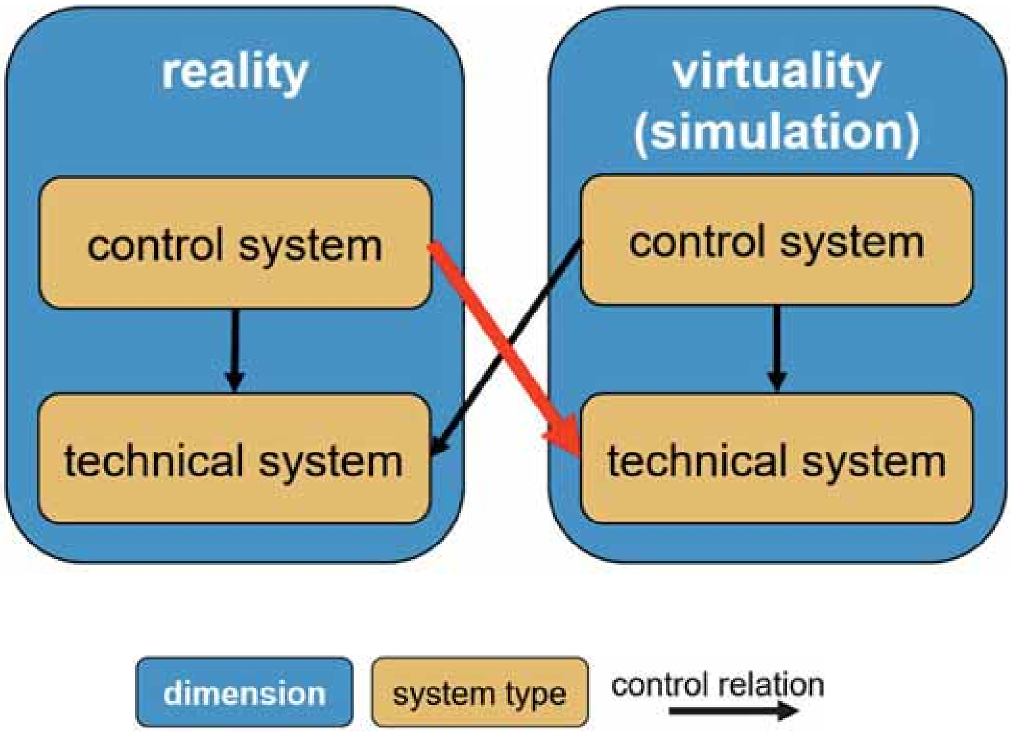
SNE SIMULATION NOTES EUROPE



Journal on Developments and Trends in Modelling and Simulation
EUROSIM Scientific Membership Journal

Vol. 35 No.2, June 2025 ISSN Online 2306-0271 DOI 10.11128/sne.35.2.1073

ARGESIM

ASIM Fachtagung

24. bis 26. Sept. 2025

2025

„Haus der Kirche“

Dresden

Schlüsselrolle Simulation: Wandel gestalten.
Herausforderungen meistern.

21. ASIM Fachtagung „Simulation in Produktion und Logistik“

Als größte europäische Tagung zum Thema Simulation im Bereich Produktion und Logistik gibt die ASIM Fachtagung alle zwei Jahre einen Überblick der zukunftsweisenden Trends, aktuellen Entwicklungen und erfolgreichen Projekte. Präsentiert und diskutiert werden wissenschaftliche Arbeiten sowie interessante Anwendungen aus der Industrie.

Themen der Tagung

Supply Chain Simulation & Logistik
- Produktionsnetzwerke & -logistik
- Intralogistik; Lieferketten
- Transport & Verkehr
- Innovative Materialflusstechnik
- Produktionsplanung & -steuerung
- Systemresilienz; (Cyber-)Sicherheit

Simulationsanwendungen in Industrie und Dienstleistung
- Automobilindustrie
- Maschinen- & Anlagenbau
- Halbleiterindustrie
- Simulation as a Service
- Energieeffizienz; Nachhaltigkeit

Methoden, Werkzeuge & Simulationstechnik
- Virtuelle Inbetriebnahme; Digitaler Zwilling; Digitaler Schatten
- Simulationsbasierte Optimierung
- Verifikation & Validierung
- Data Science; Visual Analytics; Virtual Reality; Augmented Reality
- Künstliche Intelligenz; Maschinelles Lernen
- Ereignisdiskrete Simulation; Agentenbasierte Simulation
- Interoperabilität; verteilte Simulation; Cloud-basierte Simulation

Teilnahme

Ausstellung

Die begleitende Ausstellung zeigt den aktuellen Stand von Simulationswerkzeugen, verwandter Software und Dienstleistungen im Simulationsumfeld. Falls Sie sich als Aussteller auf der Fachtagung präsentieren möchten, kontaktieren Sie uns bitte.

Teilnahmegebühr (pro Person, inkl. Mehrwertsteuer)

Im Preis enthalten sind die Teilnahme an Vorträgen, der Ausstellung, Mittagessen, Erfrischungsgetränke mit Pausensnack, ein Get-Together sowie eine Abendveranstaltung. Ein gedruckter Tagungsband kann gegen einen Aufpreis von 50,00 € optional erworben werden.

Grundpreis		1190,00 €
Frühbucherrabatt (bis 01.07.2025)	abzgl.	50,00 €
Mitglieder von ASIM, GI, EUROSIM	abzgl.	140,00 €
Einreichung Beitrag (1 x pro Beitrag und Autor)	abzgl.	170,00 €
ab dem 2. Teilnehmer einer Institution	abzgl.	140,00 €

Hochschulangehörige erhalten 40 % Nachlass auf den Endpreis.

Tagungsleitung

Prof. Dr.-Ing. habil.
Thorsten Schmidt

Professur für
Technische Logistik
Fak. Maschinenwesen
TU Dresden

Tagungsort

„Haus der Kirche“
Hauptstraße 23
01097 Dresden

Anmeldung

Alle Informationen zur Anmeldung finden Sie auf unserer Website:

[https://asim-gi.org/spl2025]

GI · ASIM · ASIM Fachtagung · Institut für Technische Logistik und Arbeitssysteme · TECHNISCHE UNIVERSITÄT DRESDEN

Editorial

Dear Readers, This second issue of SNE Volume 35, 2025, SNE 35(2), presents six contributions, showing the really broad area of modelling and simulation, as e.g. with the submitted contribution «Process Model for Integration of Speech Recognition and Understanding in Multiple Remote Tower» by O. Ohneiser et al. Five contributions in this issue continue the postconference publications from ASIM's Symposium Simulation Technique in Munich, September 2024, and start the postconference publications from ASIM's GMMS/STS Workshop 2025 at DLR Oberpfaffenhofen. Here, the contents are ranging from simulation-enhanced action-oriented process mining and development of a digital twin for a mobile articulated gripper robot via filtering effects on simulated signals with entropy methods and coupling simulation and machine learning in supply chains to simulation of pneumatics network using the DLR ThermoFluidStream Library.

For postconference publication, SNE has implemented a clearer classification. In a paragraph «Publication Remark», just before the references, SNE gives a citation of conference version and indicates the change in SNE's postconference publication with 'Improved' (correction of typos, reformatting, slight text changes), and 'Extended' ('Improved' and significant content extensions), and additionally by 'English Version'). Furthermore, I would like to remind on the also clarified licensing strategy of SNE: SNE is now licensed under Creative Commons Licence CC BY 4.0. So the basic version of SNE and all SNE contributions are published with open access. For members of the societies we still additionally provide a member version (prev. 'restricted access', now 'member access') with advanced features (colour, high-res, and eventually with references to slides, etc.)

I would like to thank all authors for their contributions, and many thanks to the SNE Editorial Office for layout, typesetting, preparations for printing, electronic publishing, and much more. And have a look at the info on EUROSIM-related simulation events of the year 2025: ASIM Conference on Simulation in Production and Logistics in Dresden, I3M 2025 conference in Fes, Morocco, WinterSim 2025 in Seattle, and further conferences and workshops of the EUROSIM societies.

Felix Breitenecker, SNE Editor-in-Chief, eic@sne-journal.org; felix.breitenecker@tuwien.ac.at

Contents SNE 35(2)

e-SNE 35(2), DOI 10.11128/sne.35.2.1073

www.sne-journal.org/sne-volumes/volume-35

SNE Basic e-Version with Open Access, CC BY 4.0

SNE Full e-Version for Members of publication-active EUROSIM Societies: ASIM, CEA-SMSG, CSSS, DBSS, KA-SIM, LIOPHANT, NSSM, PTSK, SIMS, SLOSIM, UKSIM

print-SNE for members on demand (printer INTU TU Wien)

Cover: Figure 1 (ref [4]) from contribution Özkul et al.

SNE Contact & Info

SNE Online ISSN 2306-0271, SNE Print ISSN 2305-9974

→ *www.sne-journal.org*

office@sne-journal.org, eic@sne-journal.org

SNE Editorial Office
Felix Breitenecker (Organisation, Author Mentoring)
Irmgard Husinsky (Web, Electronic Publishing),
ARGESIM/Math. Modelling & Simulation Group,
Inst. of Analysis and Scientific Computing, TU Wien
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria

SNE Simulation Notes Europe

WEB: → *www.sne-journal.org*, DOI prefix 10.11128/sne

Scope: Developments and trends in modelling and simulation in various areas and in application and theory; comparative studies and benchmarks (documentation of ARGESIM Benchmarks on modelling approaches and simulation implementations); modelling and simulation in and for education, simulation-based e-learning; society information and membership information for EUROSIM members (Federation of European Simulation Societies and Groups).

Editor-in-Chief: Felix Breitenecker, TU Wien, Math. Modelling Group
Felix.Breitenecker@tuwien.ac.at, *eic@sne-journal.org*

Print SNE: INTU (TU Wien), Wiedner Hauptstrasse 8-10, 1040, Vienna, Austria – *www.intu.at*

ARGESIM Publisher: Arbeitsgemeinschaft Simulation News c/o Math. Modelling and Simulation Group, TU Wien / 101, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria; *www.argesim.org*, *info@argesim.org* on behalf of ASIM *www.asim-gi.org* and EUROSIM → *www.eurosim.info*

SNE - Aims and Scope

Simulation Notes Europe (SNE) provides an international, high-quality forum for presentation of new ideas and approaches in simulation - from modelling to experiment analysis, from implementation to verification, from validation to identification, from numerics to visualisation (www.sne-journal.org).

SNE seeks to serve scientists, researchers, developers and users of the simulation process across a variety of theoretical and applied fields in pursuit of novel ideas in simulation. **SNE** follows the recent developments and trends of modelling and simulation in new and/or joining areas, as complex systems and big data. **SNE** puts special emphasis on the overall view in simulation, and on comparative investigations, as benchmarks and comparisons in methodology and application. For this purpose, **SNE** documents the **ARGESIM Benchmarks** on *Modelling Approaches and Simulation Implementations* with publication of definitions, solutions and discussions. **SNE** welcomes also contributions in education in/for/with simulation.

SNE is the scientific membership journal of **EUROSIM**, the *Federation of European Simulation Societies and Simulation Groups* (www.eurosim.info), also providing Postconference publication for events of the member societies. **SNE**, primarily an electronic journal **e-SNE** (ISSN 2306-0271), follows an open access strategy, with free download in basic version (B/W, low resolution graphics). Members of most **EUROSIM** societies are entitled to download **e-SNE** in an elaborate full version (colour, high resolution graphics), and to access additional sources of benchmark publications, model sources, etc. (via group login of the society), **print-SNE** (ISSN 2305-9974) is available for specific groups of **EUROSIM** societies.

SNE is published by **ARGESIM** (www.argesim.org) on mandate of **EUROSIM** and **ASIM** (www.asim-gi.org), the German simulation society. **SNE** is DOI indexed with prefix 10.11128.

Author's Info. Individual submissions of scientific papers are welcome, as well as post-conference publications of contributions from conferences of **EUROSIM** societies. **SNE** welcomes special issues, either dedicated to special areas and/or new developments, or on occasion of events as conferences and workshops with special emphasis.

Authors are invited to submit contributions which have not been published and have not being considered for publication elsewhere to the **SNE** Editorial Office.

SNE distinguishes different types of contributions (*Notes*), i.e.

- **TN** Technical Note, 6–10 p.
- **SN** Short Note, max. 5 p.
- **SW** Software Note , 4–6 p.
- **BN** Benchmark Note, 2–10 p.
- **ON** Overview Note – only upon invitation, up to 14 p.
- **EN** Education Note, 6–8 p.
- **PN** Project Note 6–8 p.
- **STN** Student Note, 4-6 p., on supervisor's recommendation
- **EBN** Educational Benchmark Note, 4–10 p.

Further info and templates (doc, tex) at **SNE**'s website, or from the Editor-in-Chief

www.sne-journal.org

office@sne-journal.org, *eic@sne-journal.org*

SNE Editorial Board

SNE - Simulation Notes Europe is advised and supervised by an international scientific editorial board. This board is taking care on peer reviewing of submission to SNE (and extended for special issues and Postconference publication):

Simulation-enhanced Action-oriented Process Mining in Production and Logistics

Felix Özkul*, Robin Sutherland, Sigrid Wenzel

Department Organization of Production and Factory Planning (pfp), University of Kassel, Kurt-Wolters-Str. 3, 34125 Kassel, Germany; **felix.oezkul@uni-kassel.de*

SNE 35(2), 2025, 71-81, DOI: 10.11128/sne.35.tn.10731
Selected ASIM SST 2024 Postconf. Publication: 2024-12-10
Received Improved: 2025-03-10; Accepted: 2025-03-20
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. The following paper was originally published at the 27th ASIM Symposium Simulationstechnik. This version includes an extended introduction and outlook. Process mining is increasingly being used to gain insights into processes based on operational data. Recently, approaches have been researched as to how these findings can be automatically transferred into process-regulating actions during system operation to correct deviations between the actual and target process in real time. However, the implementation of such action-oriented process mining mechanisms requires sufficient testing of the implemented actions in the application to prevent undesirable side effects in the real system. This article explains how discrete-event simulation in production and logistics can be used to mitigate risks in the context of implementing action-oriented process mining through the use of an emulation model. For this purpose, we present simulation-enhanced action-oriented process mining as well as a proof-of-concept implementation based on a use case.

Introduction

Discrete-event simulation (DES) has proven itself across many industries as a planning tool for production and logistics systems (e.g., see [1-2]). Its application enables users to analyze "what-if" or "how-to-achieve" questions by executing simulation experiments during the planning phase of a production or logistics system. The utilization of DES during commissioning or system operation is less mature, but the potential benefits of using DES during these system life cycle phases are also apparent (see [3]).

In the following, a distinction is made between the levels of the technical system and the control system with respect to systems in production and logistics. This distinction can also be made with appropriate modeling in the simulation application, so that there is a real technical system, a real control system, a simulated technical system, and a simulated control system.

In the context of this paper, emulation describes the use of a real control system being used in a simulated technical system (other types of coupling are described in [4]). The simulated technical system thus receives the same inputs that the emulated real technical system would receive, and the real control system can thus be tested simulatively. The following Figure 1 illustrates this relationship.

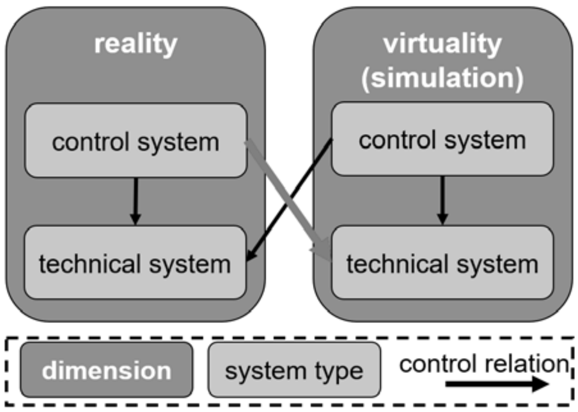


Figure 1: Coupling types of real & virtual systems [4].

DES can be used in combination with other digital methods such as process mining, for example, to reduce the effort involved in creating simulation models, or to simulatively generate synthetic input data for the application of process mining techniques [5]. The added value of the combined use of both methods has been presented in the pertinent scientific literature – also for the area of production and logistics [6].

An example in this context can be found in Özkul et al. [7] who present an extension of a holistic framework for the verification and validation of simulation models [8] that is based on the application of process mining techniques. In the broader scope of combining DES and process mining in production and logistics, the process-data-driven generation and operation of digital twins of material flow systems is also a pertinent area of research and application. Lugaresi [9] provides an overview on the state of the art and current research challenges therein ([10-11] offer a more general elaboration on the term digital twin). In the context of this very specific class of digital twins, simulation models are typically used to simulate the behavior and state of the corresponding material flow system in real time, while state information about the modeled material flow system is obtained through the application of process mining algorithms on event data that is generated in the real system.

Examples of research include the framework for generating digital twins for manufacturing systems given by Friedrich et al. [12] which exploits process mining techniques to partially automate the creation of the required simulation model; further, Overbeck et al. [13] present a method for the detection of state and structural changes during the operation of the online simulation model to automatically update and validate the model state and structure. Another recent approach is presented by Lugaresi and Matta [14], which includes a proprietary graph-oriented simulation modeling approach that utilizes object-centric process mining [15] to obtain structural information about the simulated material flow system. This approach avoids problems related to information loss, e.g., whenever a case identifier changes which typically happens during (dis)assembly processes in material flow systems.

Process mining is a research area at the intersection of data science and process science [16]. It is mainly used to analyze processes based on past operational data (see [17]) and its use is favored by the increasing prevalence of information technology systems in production and logistics [6]. The classic types of process mining are process discovery, conformance checking (see Section 1), process enhancement, and more recently, performance analysis, comparative process mining, predictive process mining (machine learning-supported process mining), and action-oriented process mining (see Section 1; [16]). These types of process mining have some overlaps in terms of both methodology and application.

In combination with DES, the combined use of both methods can be used to gain insight into both the past and the future. Not part of this given time continuum is the present and the associated consideration of the combined use of methods during ongoing system operation. Input data for process mining techniques is conventionally stored in so-called event logs (see Section 1 for more details). The process information stored in these event logs can be used, for example, to extract models describing process behavior (process discovery), and to retrospectively identify deviations between the target and actual process (conformance checking). However, for process mining to be used in an action-oriented manner in the current system operation, the process-related insight must be gained in *real time*, i.e., possibly before a process instance is completed and thus before the complete event log is generated. Streaming process mining methods (alternatively online process mining) are suitable for this purpose (see [18]). Streaming process mining analyzes so-called event streams (see Section 1) instead of conventional event logs [18], thereby gaining knowledge about process instances even before they are completed and, if necessary, enabling actions that regulate process instances.

Action-oriented process mining aims to generate these diagnostic-based actions (see [16]), thereby bridging the gap between insight and action that conventional process mining cannot. However, the implementation of such action-oriented process mining-based process control mechanisms has not yet been researched in detail in the area of production and logistics, and the risks associated with the implementation of action-oriented process mining are obvious.

For example, in case of failure, testing on the real system may affect its operational performance (e.g., due to unforeseen system failures caused by actions of the action-oriented process mining mechanism). DES could, on the one hand, help mitigate these risks by generating event streams for different process scenarios, which are used as input resources for action-oriented process mining to analyze the different response actions (e.g., an online adjustment of machine parameters or job scheduling) as well as their effects.

On the other hand, as an element of decision support, DES can help to determine the impact of non-conformity with regard to process instances to gain predictive insights into the necessary process control actions.

This enables users to mitigate the risks associated with the implementation of action-oriented process mining and helps increasing its effectiveness in practice. Our approach uses an executable emulation model of an underlying technical system to test an action-oriented process mining mechanism.

The paper is structured as follows: Section 1 explains the key process mining terms. Section 2 specifies theoretical scenarios for simulation-enhanced action-oriented process mining. Section 3 presents a use case based on a proof-of-concept implementation. The paper concludes with a summary and a research outlook.

1 Process Mining Terms and Context

In the context of this paper, a *process* refers to "a coherent series of changes that unfold over time and occur at multiple levels" ([19], p. 3). These changes are triggered by *events*. Event logs record the execution of processes based on events that start or end *activities*. Process and activity executions represent so-called instances. A process instance is also referred to as a *case*. Each case can be described by a sequence of events or activities, which is logged as a trace in the event log. In other words, event logs are a multiset of traces, with each trace being a sequence of events ([16], def. 3).

Events are assigned to their respective cases using a corresponding case identifier. Events and traces can be described in more detail using additional information (such as resources performing activities). In practical use, event logs log a finite number of events. Different types of process mining can then be applied on the basis of these logged events. Processes (as defined above) can be mapped and analyzed using process models (e.g., models in Petri net notation [20]).

Conformance checking can be used to check how well a process model is able to replay the process behavior observed in an event log. The scientific literature mentions various conformance checking methods, such as footprints, token-based replay (for Petri nets), and alignments [21]. The ability of a process model to reproduce recorded process behavior is referred to as its *fitness* [17][21] and is the most important indicator for describing process model quality (the other process model quality indicators are *precision*, *generalization*, and *simplicity*, see [21]).

On the one hand, conformance checking can be used to quantify the quality of a manually built or extracted process model in relation to an event log (i.e., recorded process behavior).

On the other hand, it is also possible to analyze the conformity of the logged cases in relation to a normative process model (i.e., a binding model specifying the target process). This approach of conformance checking is central to the implementation of simulation-enhanced action-oriented process mining.

1.1 Streaming Process Mining

The examination of completed cases using event logs enables an *a posteriori* determination of conformity in relation to a normative process model. However, for a process instance to be corrected during its execution in a production and logistics system the conformity assessment must be carried out *a priori* with regard to the completion of the process instance. This requirement can be met by analyzing event streams.

Van Zelst et al. ([22], p. 7) define an *event stream* as "a continuous stream of events executed in context of an underlying business process". An event stream comprises a – potentially infinite – number of events (as defined in Section 1). The investigation of event streams is the subject of streaming process mining and includes streaming process discovery and online conformance checking [18]. Event stream analysis allows for the processing of very large event logs (like event logs too large to be stored in memory) or for continuous monitoring of processes. The latter is important for our contribution in the context of production and logistics.

In principle, event streams can be analyzed using conventional process mining algorithms by combining events into batches and then processing them like event logs. However, there are also dedicated algorithms that enable online analyses (e.g., prefix alignments [23] or the analysis of behavioral patterns ([24], see Section 2) for conformance checking).

Van Zelst et al. [22] present an architecture (S-BAR) for the application of common process discovery algorithms in online settings using abstract representations. These techniques focus on the investigation of activity relationships and control flow. In addition, Stertz et al. [25] present an approach that analyzes the temporal relationship between activities using a temporal profile.

A temporal profile contains information about the (stochastically) expected durations of activities as well as their temporal distance and a normative process model can be infused with it to allow the application of temporal conformance checking (see Figure 2 "temporal items"). A temporal profile can, for example, be extracted from a normative process execution log (i.e., a log containing valid traces of a target process) which contains information about the start and end times of activities (i.e., start and end events) or on the basis of expert domain knowledge.

New events are compared with a temporal profile and time-related outliers are detected by calculating their Z-score [26]. Temporal conformance can also be checked in the event stream in addition to the control flow view.

The approach in this paper checks event streams based on behavioral conformance (see [24]) and temporal conformance [25] to monitor process execution online.

1.2 Action-oriented Process Mining

Streaming conformance checking based on event streams enables online monitoring of processes. Action-oriented process mining builds on this and uses streaming process mining to automatically generate actions based on detected deviations from target processes (i.e., non-conformities) that imply risks associated with the violation of process *constraints*.

In the context of production and logistics, constraints can be, for example, case-specific production deadlines that are about to be missed (violation of temporal conformance) or the prescribed production sequences that are not being followed in the ongoing process (violation of behavioral conformance).

Park and Van der Aalst [27] provide a comprehensive framework for implementing action-oriented process mining for business processes with a focus on operational support (more in [17]).

During the execution of business processes events are logged in a real information system and an event stream is generated which is then analyzed by the *constraint monitor* [27]. The constraint monitor analyzes the event stream given the abovementioned constraints (which, in our approach, are temporal and behavioral constraints), that are formalized in a constraint formula (e.g., in a production setting, one such formula could evaluate if a product is manufactured given a prescribed production sequence).

The analysis of events given the constraint formulae yields constraint instances which are then sent as a constraint instance stream to the *action controller* [27]. The action controller evaluates the incoming constraint instances and generates an *action* based on the evaluation result and given an action formula [27]. The action formula takes in the constraint instance stream and a time window to produce a set of *transactions* (i.e., operations which the underlying information system can perform). An example of such an action in a production setting would be to increase an order's priority if production delays (i.e., temporal non-conformance) are diagnosed.

This paper builds on the framework of Park and Van der Aalst [27] and extends it with simulation-related components focusing on the implementation of action-oriented process mining in production and logistics.

Drieschner et al. [28] present an approach that focusses on simulation as a learning tool for action-oriented process mining. The similarity between this approach and our contribution is that both approaches use simulation as a data generator for process mining (see Section 2). However, the focus of Drieschner et al. [28] is pedagogical and focuses on user interaction. Our approach focuses on the automated testing of an implemented action-oriented process mining mechanism. For this purpose, the simulation model emulates the real system and simulation runs are conducted.

2 Simulation-enhanced Action-oriented Process Mining

Based on the framework of Park and Van der Aalst [27], the following Figure 2 shows a possible architecture for simulation-enhanced action-oriented process mining. During the execution of a simulation model, a simulation-generated event stream is created whose events are recorded by an event monitor and evaluated with regard to their temporal and behavioral conformance.

For this purpose, the approach of Stertz et al. [25] is applied on the basis of a normative temporal profile and the formulation of temporal constraints to identify temporal outliers within the event stream. Temporal constraints can be formulated, for example, by target dates that are set for the completion of an order. If, for example, a processing activity occurs too late or takes too long, this deviation is recognized via temporal conformance checking. In addition, the event monitor monitors the behavioral conformance of incoming events.

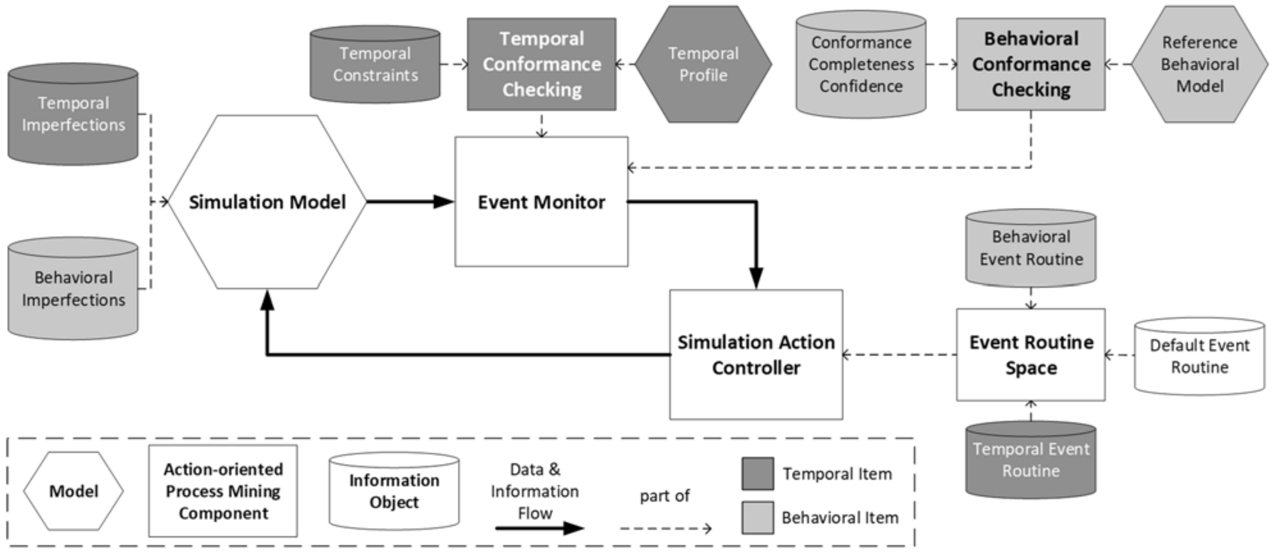


Figure 2: Architecture of the presented approach.

The behavioral conformance is assessed by analyzing *behavioral patterns* (i.e., a set of activities and possible control flow relations [24]) and a reference behavioral model (expressed as the set of *prescribed* behavioral patterns which are expected for the underlying process (see [24]). This is shown as the reference behavioral model in Figure 2. This granular perspective on the control-flow allows an in-vivo analysis of singular observable behavioral patterns (which can be inferred from events but at a higher level of abstraction) during process execution and allows for the calculation of three distinct key indicators (see [24]): C*onformance* (indicating the correctness of the observed behavior), *completeness* (indicating case progression), and *confidence* (indicating the expected stability of the conformance).

Checked events are appended to the checked event stream and relayed to the simulation action controller. Depending on the event status (i.e., whether an event conforms or not), a suitable event routine is selected from the event routine space and communicated to the simulation model. Since actions in the (discrete-event) simulation model are generated by executing event routines, we use the term *event routine space* instead of the action space (cf. [27]). We refer to the *simulation action controller* as the component that fulfils the tasks of the action controller (see [27]). If specific behavioral or temporal non-conformity is detected for an event, a routine for handling these deviating events is automatically selected (if an event conforms, the default event routine is executed). The described information flow is shown in Figure 3.

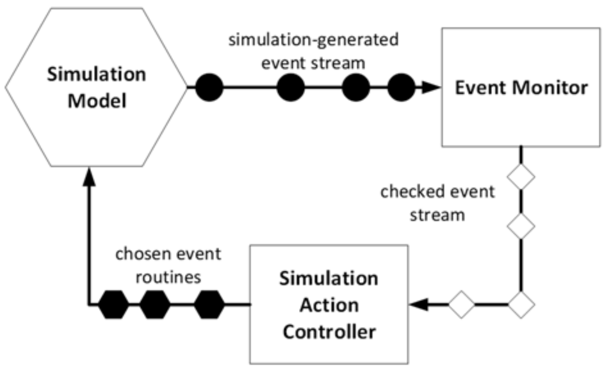


Figure 3: Information flow of the outlined approach.

The effects of the decisions by the simulation action controller can then be observed during the remainder of the simulation runs. Based on the simulation model configuration, event streams containing non-conforming events, i.e., *temporal or behavioral imperfections*, can be generated during a simulation run to invoke automatic actions of the simulation action controller, the effects of which are assessed in the simulation.

This allows the action controller to be thoroughly tested simulatively before it is implemented in the real system. Risks relating to the impairment of process sequences in the real system can be mitigated in this way. The results of the simulation application can be statistically validated through systematic experimental design and, in the context of the implementation of action-oriented process mining, should also capture rare events and faults that are difficult to observe in real system operation.

The specific number of simulation runs depends on the specific use case and the number of different scenarios which have to be addressed by the simulation action controller. This ensures that the system behavior for these edge cases is also considered in the action controller of the real system. This is the qualitative added value of the proposed approach: Based on the execution of simulation experiments that cover the range of expected cases of process deviations in the real system, action-oriented process mining can be implemented without having to make any changes to the real system.

Instead, the simulation is used as an emulation model for the real control system and risks associated with the action-oriented process mining implementation are mitigated.

3 Use Case

The following use case demonstrates the idea behind simulation-enhanced action-oriented process mining (see Section 2) using a practical case study and implements the architecture in Figure 2.

3.1 Application System

The application system is a conveyor system on a university laboratory scale, Figure 4 shows the corresponding simulation model. Starting from the source conveyor, load units are fed onto the main conveyor. Load units carry objects that have an object type that determines the target production sequence on the machines M1-M6 and one load unit corresponds to one case.

In addition, there are various stopping points (H1-H7) on the main conveyor, which can read and write to RFID tags attached to the load units. In addition to the conveyed object type, other load unit-related information (attributes) is also stored on the RFID tags, such as the priority of the order (high/low), the target time for completion of the order and the next machine in the object type-specific processing sequence.

The machines are located on side conveyors, which are connected to the main conveyor via conveyor switches. Side conveyors have a certain load unit capacity and load units can only be discharged onto a side conveyor if it can accommodate further load units.

At the conveyor switches, load units with high priority have right of way; for load units with the same priority, first in first out (FIFO) applies.

We formalize a workflow net (a Petri net with certain properties, see [17]), which is a simplified process model for the behavior of the load units (cases) in the application system (Figure 5). The normative workflow net exhibits maximum fitness in relation to the underlying conveyor system.

However, it enables additional behavior that is not desirable in the real system due to the object type-specific processing sequences (i.e., its precision is low). To increase the precision of the model, the formalization of color sets and the introduction of transition guards would be appropriate.

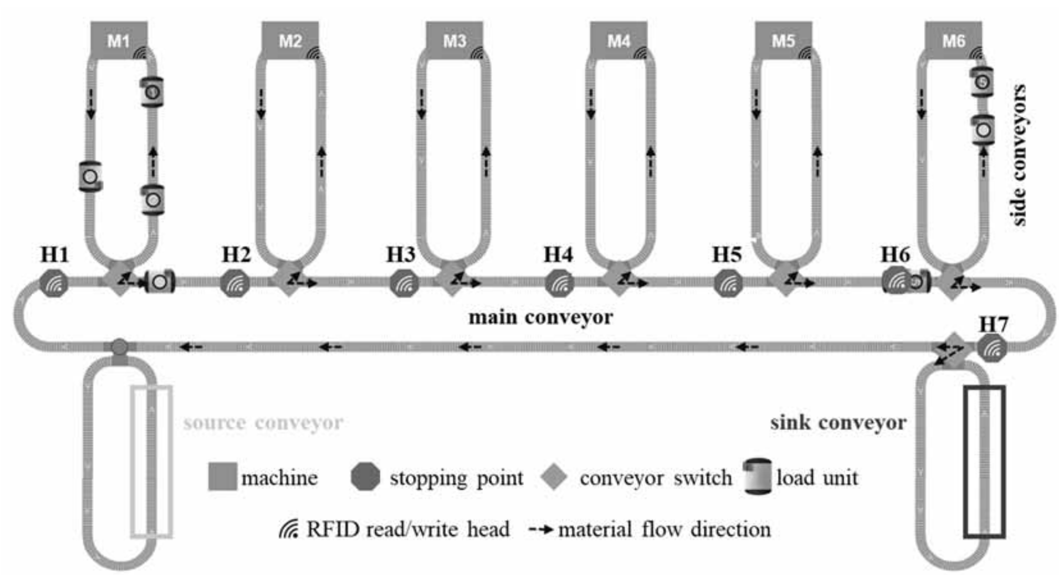


Figure 4: Simulation model of the laboratory system.

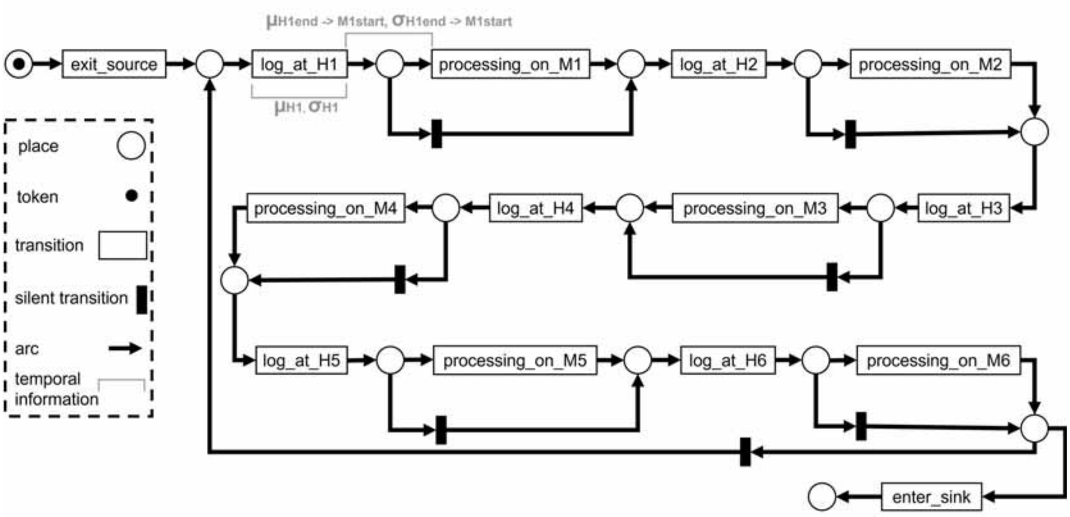


Figure 5: Workflow-net of the load unit process.

For simplicity, however, we decided not to include a colored Petri net of the load unit process. Starting from the source (exit_source), the various stopping points on the main conveyor are controlled and logging events are fired (log_at_H*).

Depending on the conveyed object type on the load unit, its processing progress and the availability of side conveyor capacity, the load units are processed at the machines (processing_on_M*) or conveyed in a waiting loop. Skips are modeled as silent transitions; load units that cannot be processed at their destination machine will still pass the subsequent stopping points on the main conveyor (i.e., it is not possible to skip logging activities, which is coherent with respect to the layout in Figure 4).

Furthermore, the process model is infused with a temporal profile, which is based upon domain knowledge about the controls of the reference system. For clarity, we graphically omit the temporal distances between all activities and activity (transition) durations (examples for both are highlighted in blue in Figure 5).

3.2 Use Case

The goal of the use case is to illustrate how the approach can handle incorrect processing sequences (behavioral non-conformance) and processing delays (temporal non-conformance) and how automated actions can address them. Due to the stochastic order loading of different object types, congestion of load units can occur in the system, which delays the processing of orders.

Furthermore, it is assumed that the reading/writing of the RFID tags – which are placed on the load units – is not always error-free, and that read and write errors can occur. A case is considered compliant if its processing sequence corresponds to the target processing sequence of its object type and the order is completed in time. If the target processing sequence is violated, the event monitor detects this deviation as behavioral non-conformance. Delays related to the processing of load units may lead to the detection of temporal non-conformance (see Section 2).

In case of behavioral non-conformance, rerouting to the original target machine is defined as an automated action in the event routine space of the simulation action controller.

In case of temporal non-conformance, the priority of a load unit must be increased if there is a delay so that it can be routed to the conveyor switches more quickly if necessary. In the event of temporal non-conformance of a processing activity, it must also be assumed that a machine requires maintenance. A maintenance worker should therefore be sent out in this case to carry out the maintenance as quickly as possible.

3.3 Technical Implementation

We implement the architecture from Figure 2 using AnyLogic 8 and open-source process mining software systems.
Figure 6 shows the proof-of-concept implementation.

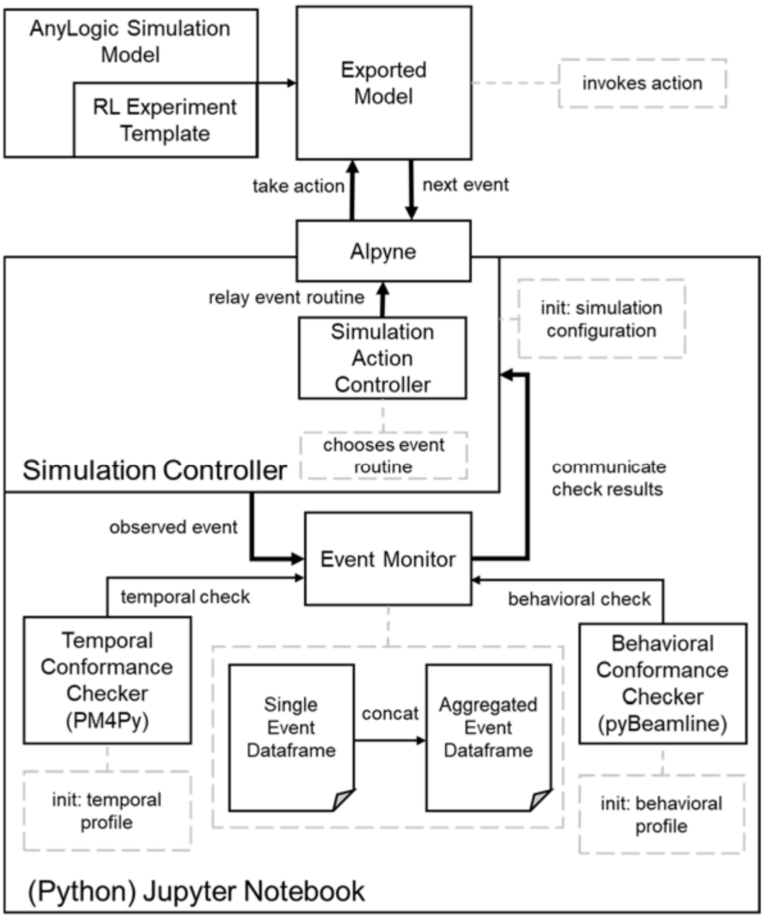


Figure 6: Structure of the proof-of-concept implementation.

First, a simulation model is built and an experiment template is implemented. The current AnyLogic software ecosystem provides a template for Reinforcement Learning (RL) (i.e., the possibility to define observations, actions, and configurations) which can be adapted because – similarly to RL – our approach utilizes the observation and action capabilities in the simulation context.

Therefore, we repurpose the RL pipeline to implement simulation-enhanced action-oriented process mining. The simulation model containing the experiment template is exported to a standalone (Java) model, which is bi-directionally linked to a Python Jupyter Notebook using the Alpyne software library [29]. The exported model is configured to invoke actions based on certain key events, those being the logging events shown in Figure 5 at the stopping points (H1-H6).

Whenever a logging event occurs, the simulation engine pauses the simulation run and passes the events to the Python simulation controller, which implements Alpyne as its simulation interface.

The observed event is then relayed to the event monitor which transforms and logs the observed event in a Pandas Dataframe object (Single Event Dataframe) to perform temporal and behavioral conformance checking (https://pandas.pydata.org/docs/). Software systems for the implementation of streaming conformance checking (the central component of the event monitor) are currently only available to a limited extent (see [30]). The comprehensive Python-based open-source framework PM4Py [31] implements the temporal conformance checking approach of Stertz et al. [25].

Burattin [30] provides an open-source streaming process mining software named Beamline, which is available for Java and Python (pyBeamline). Our work implements pyBeamline's behavioral conformance checking [24][30] and provides a compact implementation. Based on a reference event log (which can be generated, for example, by simulating the workflow net shown in Figure 5 or by exporting traces from a valid simulation model after the warm-up phase) a normative behavioral model is initially mined with pyBeamline and new event instances are checked against it.

In order to assess the behavioral conformance of a running case, we implement a second Dataframe object to which each event is appended (technically, at each iteration the Single Event Dataframe and the Aggregated Event Dataframe are concatenated). This step is computationally costly but necessary to contextualize events with regard to the previous events of their running case. Furthermore, each event is checked against the temporal profile of the normative process which is created using PM4Py (the computation of a temporal profile requires start and end timestamps for activities). The PM4Py implementation additionally requires floating point numbers as timestamps whereas other algorithms often work based on datetime formatted timestamps. After conformance checking, the results of the check are relayed to the simulation action controller which then chooses an appropriate event routine (action). If the event monitor does not register non-conformance, the default routing logic is applied to the load unit (see Section 3.1).

If, for example, the event monitor detects that a case is being processed late (temporal non-conformance), the simulation action controller increases its priority to enable faster transport to the processing machines and schedules a maintenance event. After selecting a suitable action, it is communicated to the simulation model via Alpyne and the simulation engine continues the simulation run. The control logic of the simulation can be modified by the simulation controller (with appropriate implementation in the simulation model) to specifically generate non-conforming events, for example by generating misreads at the stopping points with a predetermined probability, which ensures that load units are incorrectly ejected or not ejected. In this way, the components of action-oriented process mining can be specifically checked within the framework of simulation experiments with regard to their ability to detect and recommend and thus their suitability as operational support before they are implemented in the real system (see Section 2). This avoids risks associated with undesirable side effects in reality.

4 Summary, Limitations, and Research Opportunities

This article presents a combination approach of discrete-event simulation and process mining, in which a simulation model is used as an emulation model for the implementation of action-oriented process mining. The qualitative added value lies in the mitigation of risks associated with undesirable or unexpected side effects of the automation of process-regulating actions in production and logistics systems.

Furthermore, the use of simulation allows the introduction of targeted imperfections to improve the testing of action-oriented process mining mechanisms. However, the current results can only be seen as a starting point for subsequent research challenges. These concern, among other things, the incorporation of a more dynamic event routine space (see Figure 2) and the explicit consideration of data and information quality.

Event routine space exploration. The current implementation uses a rule-based mechanism for generating event routines because the underlying use case is of lower complexity than industrial systems. Herein lies a limitation of the proposed approach. Possible research directions in this context include the exploration of the event routine space using RL (since the current software implementation is already predestined for it).

Challenges herein are – among others – the design of an application-specific reward function for the RL agent(s), the training of an agent given a vast action space in reality, and the validation of the actions taken by the agent, because their effectiveness may not always be observable.

Digital twinning. As mentioned in the Introduction, process-data-driven digital twins are an active area of research in the context of combining DES and process mining in production and logistics.

The given framework and implementation in Sections 2 and 3 utilize online process mining of event streams as opposed to traditional process mining which analyzes event logs (see Sections 1 and 1.1) on the (completed) case-level. This enables analysis and intervention on the event-level, which maximizes the real-time capabilities of the monitoring system because events are atomic in DES models – there is no smaller point of intervention. Consequently, the presented work can be seen as a technological foundation for improving the real-time capabilities of existing frameworks.

However, to achieve the right degree of real-time capabilities, research challenges include, among others, the development of a suitable simulation lookahead mechanism which performs a useful lookahead to choose suitable corrective actions (in case of non-conformance) while the underlying material flow system is in operation. This is not trivial because – depending on the specific use case (i.e., the specific real-time requirements) – there is a differing amount of time to look ahead, and one must choose a fitting (and potentially dynamic) factor between the time required by a computer to run the simulation (simulation model runtime) and the model time needed to obtain useful simulation results (i.e., how far into the future one has to look).

Information and data quality. The approach given in this work analyzes event streams containing events that describe the execution of activity instances in a case. Events have attributes such as a timestamp, an activity label, and a case identifier, which are the basis for applying process discovery and conformance checking algorithms.

Their attribute values contain the relevant information about the analyzed cases and are sufficient for the most basic types of analysis such as the discovery of process control flows and activity structures – given the right information and data quality.

In the context of applying process mining techniques, process analysis often follows a data-centric perspective wherein the focus lies on "getting mileage" out of the data available in the underlying system (e.g., data from sensors). Today, systems in production and logistics possess advanced technological data collection capabilities and collect vast possible input data for process analyses.

This typically results in the collection of unnecessary data that may not contain the information needed for a specific application scenario (even after event log processing), or data that is too vast to be purposefully analyzed, thus reducing the quality and therefore applicability of analysis results.

More advanced applications, such as those that combine DES and process mining, e.g., to perform decision point analysis for discovering simulation control strategies (see [7]), typically require additional information compared to traditional process mining techniques, which must be obtained from other information sources. To meet the resulting information quality requirements, the combination of DES and process mining should be done from an information-centric perspective, wherein the focus lies on the specific objective information need and the identification of potentially available information, before gathering the associated input data. Herein lies further research need for application support.

Publication Remark.

This contribution is the improved version of the conference version published in
Tagungsband Langbeiträge ASIM SST 2024,
ARGESIM Report AR 47, ISBN ebook: 978-3-903347-65-6, pp 193-201, DOI 10.11128/arep.47.a4717

Literature

[1] Gutenschwager K, Rabe M, Spieckermann S, Wenzel S. *Simulation in Produktion und Logistik*. 1st Edition. Berlin, Heidelberg: Springer; 2017. 290 p.

[2] Verein Deutscher Ingenieure. VDI 3633 Blatt 1 – Simulation von Logistik-, Materialfluss- und Produktionssystemen – Grundlagen. 2014, Berlin: DIN Media.

[3] Bicalho-Hoch A L, Özkul F, Wittine N, Wenzel S. A Tool-Based Approach to Assess Simulation Worthiness and Specify Sponsor Needs for SMEs. In Feng B, Pedrielli G, Peng Y, Shashaani S, Song E, Corlu CG, Lee LH, Chew E P, Roeder T, Lendermann P, editors. Proc. of 2022 Winter Simulation Conference. *2022 Winter Simulation Conference (WSC)*; 2022 Dec; Singapore. Institute of Electrical and Electronics Engineers: Piscataway, New Jersey. 1818-1829. DOI 10.1109/WSC57314.2022.10015373

[4] Follert G, Trautmann A. Emulation intralogistischer Systeme. In Wenzel S, editor. Simulation in Produktion und Logistik. 12. ASIM-Fachtagung; 2023 Sep; Kassel. San Diego, Erlangen: Society for Modeling and Simulation International SCS Publishing House e.V. 521-530.

[5] Özkul F, Sutherland R, Wenzel S, Jessen U, Spieckermann S. Verknüpfung von ereignisdiskreter Simulation und Process Mining in Produktion und Logistik. In Breitenecker F, Deatcu C, Durak U, Körner A, Pawletta T, editors. ASIM SST 2022 Proceedings Langbeiträge. 26. ASIM Symposium Simulationstechnik; 2022 Jul; Vienna. ARGESIM Publisher: Vienna. 39-48. DOI 10.11128/arep.20.a2035.

[6] Spieckermann S, Stöhr N, Mayer G, Özkul F, Wenzel S. Fallbeispiele aus Produktion und Logistik für die Verknüpfung von ereignisdiskreter Simulation und Process-Mining. In Bergmann S, Feldkamp N, Souren R, Straßburger S, editors. Simulation in Produktion und Logistik. 20. ASIM-Fachtagung; 2023 Sep; Ilmenau. Ilmenau: Universitätsverlag Ilmenau. 155-165. DOI 10.22032/dbt.57785.

[7] Özkul F, Sutherland R, Wenzel S, Spieckermann S. Einsatz von Process-Mining zur Verifikation und Validierung von Simulationsmodellen in Produktion und Logistik. In Bergmann S, Feldkamp N, Souren R, Straßburger S, editors. Simulation in Produktion und Logistik. 20. ASIM-Fachtagung; 2023 Sep; Ilmenau. Ilmenau: Universitätsverlag Ilmenau. 155-165. DOI 10.22032/dbt.57784.

[8] Rabe M, Spieckermann S, Wenzel S. *Verifikation und Validierung für die Simulation in Produktion und Logistik. Vorgehensmodelle und Techniken*, Berlin: Springer Publishing; 2008. 237 p.

[9] Lugaresi G. Process Mining as Catalyst of Digital Twins for Production Systems: Challenges and Research opportunities. In Lam H, Azar E, Batur D, Gao S, Xie W, Hunter S R, Rossetti M D, editors. Proc. of the 2024 Winter Simulation Conference. *2024 Winter Simulation Conference (WSC)*; 2024 Dec; Orlando. Institute of Electrical and Electronics Engineers: Piscataway, New Jersey. 3082-3093. DOI 10.1109/WSC63780.2024.10838896.

[10] International Organization for Standardization. ISO/IEC 30173:2023 - Digital twin — Concepts and terminology. 2023, Genf: ISO Central Secretariat.

[11] Stark R, Kind S, Neumeyer S. Innovations in digital modelling for next generation manufacturing system design. *CIRP Annals*. 2017; 66: 169-172. DOI 10.1016/j.cirp.2017.04.045.

[12] Friederich J, Francis D P, Lazarova-Molnar S, Mohamed N. A framework for data-driven digital twins of smart manufacturing systems. *Computers in Industry*. 2022; 136: 103586. DOI 10.1016/j.compind.2021.103586.

[13] Overbeck L, Graves S C, Lanza G. Development and analysis of digital twins of production systems. *International Journal of Production Research*. 2024; 62: 3544-3558.
DOI 10.1080/00207543.2023.2242525

[14] Lugaresi G, Matta A. Automated digital twin generation of manufacturing systems with complex material flows: graph model completion.
Computers in Industry. 2023; 151: 103977, 1-14.
DOI 10.1016/j.compind.2023.103977

[15] Van der Aalst W M P. Object-Centric Process Mining: Dealing with Divergence and Convergence in Event Data. In: Ölveczky P C, Salaün G, editors. Software Engineering and Formal Methods. *SEFM2019*; Sep 2019; Oslo. Springer International Publishing: Cham. 3-25.
DOI doi.org/10.1007/978-3-030-30446-1_1

[16] Van der Aalst W M P. Process Mining: A 360 Degree Overview. In: Van der Aalst W M P, Carmona J, editors. *Process Mining Handbook*. 1st edition. Cham: Springer International Publishing; 2022. p 3-34.

[17] Van der Aalst W M P. *Process mining. Data science in action*. 2nd Edition. Berlin, Heidelberg: Springer Publishing; 2016. 467 p.

[18] Burattin A. Streaming Process Mining. In: Van der Aalst W M P, Carmona J, editors. *Process Mining Handbook*. 1st edition. Cham: Springer Publishing; 2022. p 349-372.

[19] vom Brocke J, Van der Aalst W M P, Grisold T, Kremser W, Mendling J, Pentland B, Recker J, Roeglinger M, Rosemann M, Weber B. Process Science: The Interdisciplinary Study of Continuous Change. *SSRN Electronic Journal*. 2021; 1-9.
DOI 10.2139/ssrn.3916817

[20] Reisig W. *Understanding Petri Nets*. 1st Edition, Berlin, Heidelberg: Springer Publishing; 2013. 236 p.

[21] Carmona J, Van Dongen B, Solti A, Weidlich M. *Conformance checking. Relating processes and models*. 1st Edition, Cham: Springer Nature; 2018. 285 p.

[22] Van Zelst S J, Van Dongen B F, Van der Aalst W M P. Event stream-based process discovery using abstract representations. *Knowledge and Information Systems*. 2017; 54: 407-435.
DOI 10.1007/s10115-017-1060-2

[23] Van Zelst S J, Bolt A, Hassani M, Van Dongen B F, Van der Aalst W M P. Online conformance checking: relating event streams to process models using prefix-alignments. *International Journal of Data Science and Analytics*. 2019; 8: 269-284.
DOI 10.1007/s41060-017-0078-6

[24] Burattin A, Van Zelst S J, Armas-Cervantes A, Van Dongen B F, Carmona J. Online Conformance Checking Using Behavioural Patterns. In Weske M, Montali M, Weber I, vom Brocke J, editors. Business Process Management. *16th International Conference on Business Process Management*; 2018 Sep; Sydney, Australia. Cham: Springer Publishing. 250-267.
DOI 10.1007/978-3-319-98648-7_15

[25] Stertz F, Mangler J, Rinderle-Ma S. Temporal Conformance Checking at Runtime based on Time-infused Process Models. 2020:
DOI 10.48550/arXiv.2008.07262

[26] Crocker L, Algina J. *Introduction to classical and modern test theory*. 2nd Edition, New York: Holt, Rinehart and Winston; 1986. 527 p.

[27] Park G, Van der Aalst W M P. Action-oriented process mining: bridging the gap between insights and actions. *Progress in Artificial Intelligence*. 2022; 1-22.
DOI 10.1007/s13748-022-00281-7

[28] Drieschner C, Heckl A, Krcmar H. Simulation Tool for Learning Action-Oriented Process Mining. In Castro M, El-Abd M, Zeid A, editors. EDUCON 2023 Conference Proc. *IEEE Global Engineering Education Conference 2023*, 2023 May; Kuwait. Institute of Electrical and Electronics Engineers: Piscataway, New Jersey. 1-8.
DOI 10.1109/EDUCON54358.2023.10125248

[29] Wolfe-Adam T, Alpyne - A Python library for executing AnyLogic models, Anylogic.
https://github.com/t-wolfeadam/Alpyne

[30] Burattin A. Beamline: A comprehensive toolkit for research and development of streaming process mining. *Software Impacts*. 2023; 17(100551):1-3.
DOI 10.1016/j.simpa.2023.100551

[31] Berti A, Van Zelst S, Schuster D. PM4Py: A process mining library for Python. *Software Impacts*. 2023 17(100556):1-7.
DOI 10.1016/j.simpa.2023.100556.

Development of a Digital Twin for a Mobile Articulated Gripper Robot in Simscape Multibody

Andreas Apostolatos[1,2*], Jan Janse van Rensburg[1], Sebastian Groß[1], Stefan Kerber[1], Claudia Rubio Hervás[2], Minas Apostolakis[2], Chi-Chia Tung[2], Miguel A. Vega Torres[2], Steve Miller[1]

[1]The MathWorks GmbH, Weihenstephaner Straße 6, 81673 Munich, Bavaria, Germany; *aapostol@mathworks.com, andreas.apostolatos@tum.de

[2]Technical University of Munich, Arcisstraße 21, 80333 Munich, Bavaria, Germany

SNE 35(2), 2025, 83-91, DOI: 10.11128/sne.35.tn.10732
Selected ASIM SST 2024 Postconf. Publication: 2024-12-10
Received Improved: 2025-03-11; Accepted: 2025-03-20
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Mobile articulated gripper arms are revolutionizing a range of fields, from additive manufacturing and packaging to automated assembly lines and surgical robotics. These innovations underscore the crucial role of simulation in design and development. Digital Twins are integral to this process, ensuring robustness, performance optimization, real-time monitoring and control, and adherence to industry standards. Preparing students for successful careers to develop such innovative engineering solutions leveraging state-of-the-art methods and approaches is of high importance. In this work, we demonstrate how *Project-Based Learning* (PBL) for such complex engineering tasks can be significantly enhanced and accelerated using MATLAB® and Simscape™ Multibody™. The results of this project clearly show that students can deliver high-end, robust solutions to complex engineering tasks, allowing them to efficiently familiarize themselves with advanced engineering topics by leveraging industry-mature modeling, simulation, test, and deployment ecosystems provided by MathWorks®.

Introduction

Today's job market requires highly skilled engineers equipped with the knowledge and tools to elaborate, develop, test, and eventually deploy qualified engineering solutions for diverse applications.

As new technologies continuously emerge, the need for sophisticated development methods and comprehensive environments increases, enabling the handling of ever broadening application fields. Therefore, modern education necessitates the adoption of effective and efficient teaching methods such as *Flipped Classroom* and *Project-Based Learning* (PBL), see also in [1, 2, 3, 4]. These allow students to gain practical experience using state-of-the-art platforms and methods they will eventually encounter in industry.

Figure 1: Kinova Gen3 robotic arm [5].

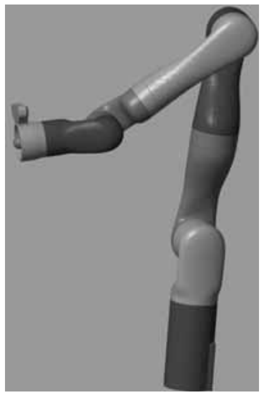


Figure 2: Kinova Gen3 robotic arm modelled in Simscape Multibody.

The *Technical University of Munich* (TUM) in Munich, Germany, is one of the leading technical academic institutes in Europe and worldwide. It offers many diverse and interdisciplinary study programs in engineering. Most these programs, if not all, involve PBL as part of their curricula.

This publication illustrates the development and investigates the results of a student project within the frame of PBL-based course "Software Lab" that took place in 2023. This lab is offered to master-level students at TUM annually. Academic and industrial partners collaborate with the participants by offering projects to pursue during this course, thus enabling students to elaborate practical projects of engineering significance in both academia and industry.

The results show that the use of PBL with industry-established engineering software increases motivation for the project elaboration, promotes learning outcomes, and enables students to develop practical and industry-relevant skills.

1 Problem Definition

This student project was conducted in collaboration with MathWorks®, the makers of MATLAB® and Simulink® modeling and simulation platforms. Simscape™ Multibody™, an add-on product to Simulink, was used as Simulation platform. Simscape Multibody enables the modeling and simulation of multibody systems, which are integral part of robotic solutions. Coupled with the Control Design and Hardware Deployment capabilities of Simulink, Simscape Multibody enables modeling, simulation, test, and deployment of complex, industrial robotic systems.

The student team's engineering task was the development of a Mobile Articulated Gripper Robotic Arm, which can be used for tasks such as additive manufacturing, surgery, search and rescue missions, etc. As starting point for the project, the Multibody model [6] of the Kinova Gen3 [5] robotic arm was provided. The actual Kinova Gen3 robotic arm is shown in Figure 1, whereas its Multibody representation in Simscape Multibody is depicted in Figure 2. It is evident that the digital model exhibits strong resemblance to its physical counterpart.

The project objectives included:

- The setup of a multibody system for the mobile base with three wheels in Simscape Multibody,
- the setup of a multibody system for the mobile base coupled with the multibody representation of the articulated Gen3 robotic arm from Kinova,
- the implementation of a path following maneuver for the mobile base,
- the implementation of a "pick and place" maneuver for the articulated robotic arm using inverse kinematics, and
- the coordination of the different tasks of the robot using a *Task Scheduler*, aiming to replicate an automated wall construction using Simscape Multibody.

The project goals were achieved in a timely manner with remarkable efficiency given the students' limited knowledge in robotics, MATLAB, and Simscape Multibody prior to the project elaboration.

The outcome of this study shows that MATLAB, Simulink, and Simscape Multibody form a strong platform for PBL due to their flexibility and adaptability. This enables students to fully leverage their skills while working on complex tasks.

Familiarization with the tools and methods needed for an engineering project like this is not trivial. The students were asked to use the MathWorks self-paced *Online Training Suite* (OTS) [7] to gain the required tools knowledge and skills. These self-paced online courses allow for an efficient and seamless familiarization with the MathWorks products and environments, necessary for the project elaboration.

2 Project Elaboration

The project was expected to be elaborated and finalized throughout an academic year. Regular presentations were scheduled to allow the project participants disseminate their results, share the project status, and respond to questions from their peers, the project supervisors, and the course coordinators.

2.1 Articulated robotic arm and gripper end effector

A multibody system in `urdf`-format for the Kinova Gen3 robotic arm was provided as a starting point for the project. Function `smimport` [8] from Simscape Multibody allows for converting a CAD model (or an assembly) to a multibody system in Simscape Multibody, which considerably accelerates the development process.

In practice, one would employ torque control to move the joints of the articulated robotic arm and gripper end effector. It was suggested that motion control should be used instead to simplify the task.

This modeling approach is suitable for early-stage design. Thus, the desirable motion was simply applied to the robotic joints. This suitable for early-stage design simplification allowed the students to complete the project promptly without the need for time-consuming controller tuning. Moreover, this modeling approach is equivalent to assuming that all necessary torque can be applied to the joints to achieve the desirable joint motion.

Subsequently, a set of waypoints was defined to have all joints of the articulated robotic arm operate in a coordinated manner so that the end effector can achieve the desirable end position. Using an appropriate number of waypoints allows the joints to find unique motion paths, leveraging the *Inverse Kinematics* (IK) solver in Simscape `simscape.multibody.KinematicsSolver` [9]. Just using a handful of waypoints might not be enough for the IK solver to find unique actuations for all joints, as the inverse kinematics problem is by nature highly non-linear.

Figure 3 shows the motion of the end effector gripper moving from the initial vertical position to finally reaching the brick on the ground. This highlights the efficiency and robustness of IK solver in Simscape.

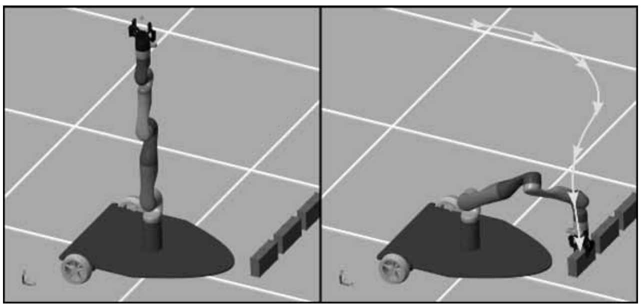


Figure 3: Inverse Kinematics Solver in Simscape (`simscape.multibody.KinematicsSolver`).

2.2 Mobile base

The students decided to use a three-wheeled model to model the mobile base. This is convenient because it provides the necessary stability and allows for steering by controlling the rotational speed of each rear wheel. Additionally, there is no need for a suspension. This design is inspired by one of the models in the SolidWorks® tutorial; see in [10] for more information.

Exporting the model from SolidWorks® to Simscape multibody can be achieved using the *Simscape Multibody Link* [11].

Various studies with different parameters were performed to find the most appropriate parameters that would allow the model to simulate stably. Moreover, the contact behavior between the ground and the wheels was modeled using the *Spatial Contact Force* block [12] in Simscape. Tables 1 and 2 summarize the values chosen for the normal and frictional contact force coefficients used in this project, respectively.

Parameter	Value
Method	Smooth Spring-Damper
Stiffness	1*e*6 N/m
Damping	1*e*4 N/(m/s)
Transition region width	0.3 m

Table 1: Normal contact behavior.

Parameter	Value
Method	Smooth Stick-Slip
Coefficient of static friction	0.9
Coefficient of dynamic friction	0.7
Critical velocity	1*e*-2 m/s

Table 2: Frictional contact behavior.

Subsequently, the students sought to impose a path the mobile base should follow. As the robotic platform under consideration is of differential drive robot type, they decided to use the *Pure Pursuit* block in Simulink, (see in [13]), to obtain the linear and angular velocity control commands necessary for navigating the mobile base via a track defined by a provided set of waypoints. The students leveraged a *MATLAB Function* block [14] to convert these linear and angular velocity control commands to the rotational speed of the left and right wheels of the mobile base. The corresponding Simulink schematic is depicted in Figure 4. The Pure Pursuit block in Simulink is provided with the robot's pose, namely, its position and heading of the robot, and the corresponding waypoints it should navigate through.

It returns the necessary control inputs for the linear and angular (rotational) velocities v and ω, respectively, that the robot must possess to navigate via the provided waypoints with the given pose.

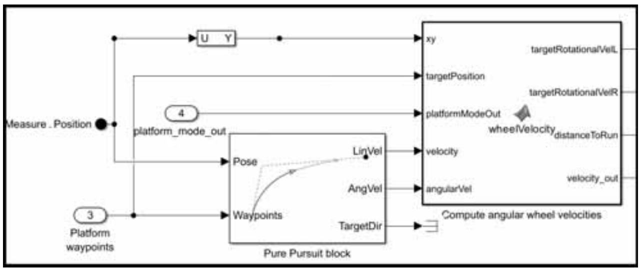


Figure 4: Pure Pursuit block and computation of the left and right wheel rotational velocities.

However, these control signals need to be converted to rotational speeds for the left and right wheel of the mobile base. To achieve that, the following formulas were employed and implemented in the MATLAB Function block with name *"Compute angular wheel velocities"*:

$$\omega_{\mathrm{L}} = \frac{v - \frac{d}{2\omega}}{r} \tag{1a}$$

$$\omega_{\mathrm{R}} = \frac{v + \frac{d}{2\omega}}{r} \tag{1b}$$

Please note that d and r in Equations 1 stand for the track width, namely, the distance between the centerline of the two rear wheels, and the radius of each rear wheel, respectively.

It is worth noting that the pure pursuit algorithm cannot stabilize the robot once it has reached its destination. For this purpose, a triangular velocity profile reduction was implemented to stabilize the behavior of the mobile base. This ensures that the mobile base reaches its destination with a gradual reduction in its velocity, ultimately stopping at the destination. A proportional controller is used to obtain the required torque that needs to be applied at the left and right wheels to achieve the desirable rotational speed. The results of the target and achieved rotational velocities for both wheels are shown in Figure 5. A three-fold scaling of the error in terms of the rotational velocity for both wheels is used for the proportional controller. This simple P-controller can almost perfectly control the torque to achieve the desirable rotational velocity.

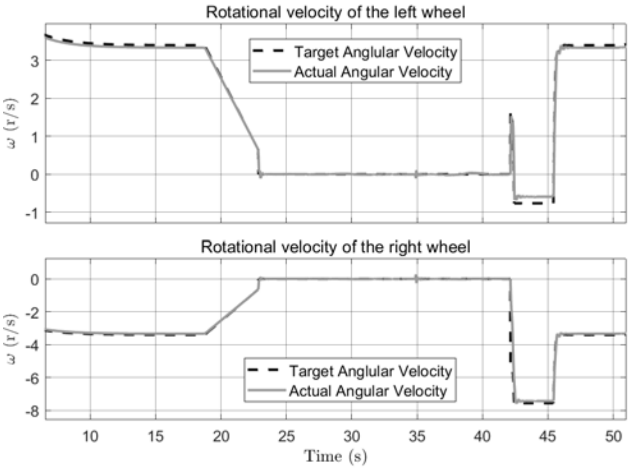


Figure 5: Target and achieved rotational velocities for both wheels of the mobile base using a proportional controller.

The triangular velocity profile employed for reducing the wheel's rotational velocity in the vicinity of the destination between can be clearly seen in Figure 5 at time instances 19 and 22.8 seconds. Employing this approach enables the mobile platform to reach its destination with stability and efficiency.

The waypoints are defined such that the mobile base navigates from the brick picking to the brick placing position. Moreover, Figure 6 depicts the track of the mobile base.

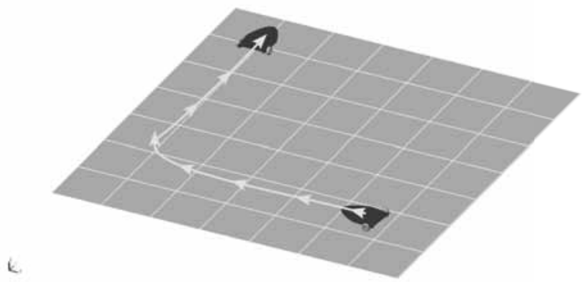


Figure 6: Mobile base navigating via waypoints using the Pure Pursuit algorithm in Simulink.

2.3 Coupling of the articulated robotic arm and the mobile base

Thus far, it has been highlighted how the students modeled and simulated the articulated robotic arm together with the gripper end effector and the mobile base separately in Simscape Multibody.

A *Weld Joint* block [15] was used to establish a rigid connection between the two; see Figure 7.

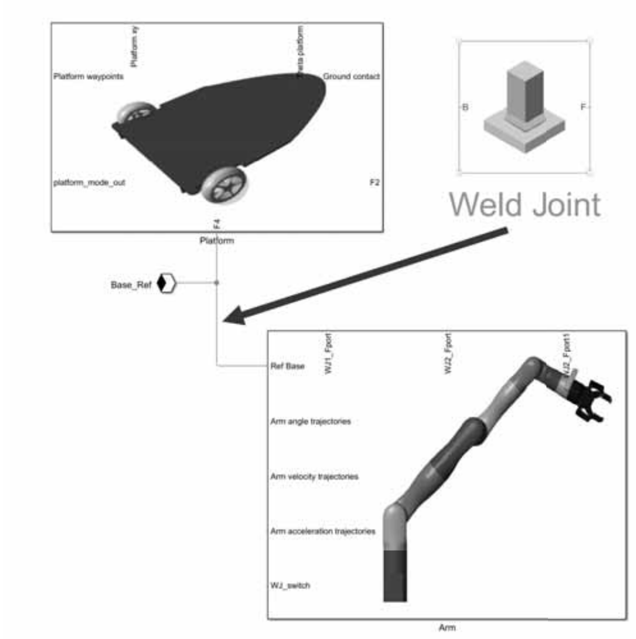


Figure 7: Rigid connection of multibody parts using the Weld Joint block in Simscape Multibody.

Joints in Simscape Multibody enable coupling different parts of a multibody system by introducing *Degrees of Freedom* (DOFs) that may restrict or enable the desirable motions depending on the application. Since the mobile base can move freely in this case, using a Weld Joint block between the mobile base of the articulated gripper arm has the desirable effect of mounting the articulated gripper arm onto the mobile base rigidly.

2.4 Power supply for the rear wheels of the mobile base

The electrical system used to power the rear wheels of the mobile base is modeled in Simscape as a DC motor. The electrical motor is connected to a battery, shown in the blue section of the Simscape model in Figure 8. The *Motor & Drive (System Level)* block [16] was used to convert the electric current from the electrical part of the Simscape model to the mechanical part (see the green part in the Simscape mode in Figure 8). Simscape offers many blocks that allow Simscape models to be interfaced from different domains (mechanical, electrical, fluid, etc.). Finally, connecting the mechanical Simscape system in green to the Simscape Multibody system is also necessary.

The torque and speed connections are denoted in Figure 8 using brown color. The *Rotational Multibody Interface* block is used [17] for this purpose.

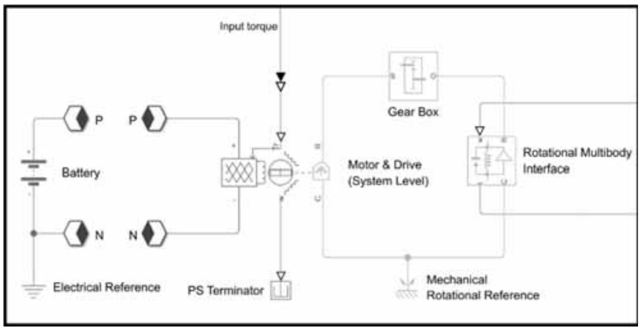


Figure 8: Modeling of a DC motor in Simscape.

The Rotational Multibody Interface block converts the mechanical force from the mechanical part of the Simscape model to torque that can be applied to the corresponding joints in the Simscape Multibody system. In this case, the torque is applied directly to the rear wheels of the mobile base. As mentioned in Section 2.2, the torque is controlled using a proportional controller by scaling the error in the rotational velocity of each of the rear wheels threefold. Each rear wheel is powered by identical DC motors, as shown in Figure 8.

2.5 Task scheduler

A task scheduler had to be implemented in the simulation model to handle task management. The mobile articulated robotic arm must drive to the location of the bricks, pick up each brick, bring the robotic arm into a vertical position (idle position) minimizing the inertia while driving, drive to the location where each brick is supposed to be placed, and finally place the brick.

The four main tasks that the task scheduler organizes include:

- Driving towards the destination where the bricks are stored,
- picking up one brick at a time,
- restoring the robot's arm posture to vertical (idle position),
- driving towards the destination where the bricks are to be placed,
- placing each brick to construct a wall.

These steps are shown in Figures 9, 10, 11, 12 respectively. The task scheduler is implemented using a MATLAB Function block by means of switch branches based on a flag. The logic of the task scheduler could be also implemented graphically using Stateflow®, but the students decided to use MATLAB code in this project instead.

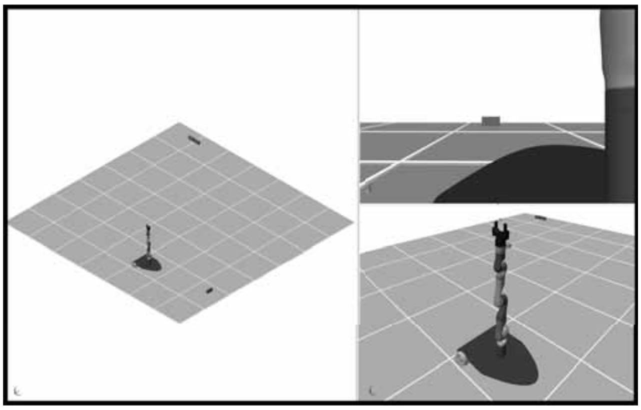


Figure 9: Task scheduler: Drive towards the destination where the bricks are stored with the arm in vertical posture (idle position).

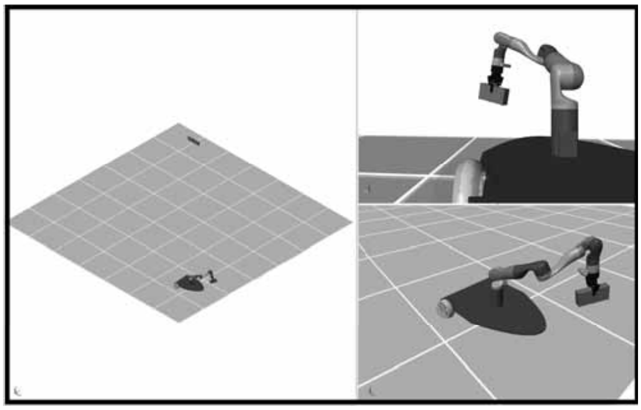


Figure 10: Task scheduler: Pick up each brick using the IK solver in Simscape.

3 Project Results and Learning Outcomes

The main project objective of modeling and simulating a mobile articulated gripper robot that can pick and place bricks to construct a wall using Simscape Multibody has been achieved with remarkable efficiency.

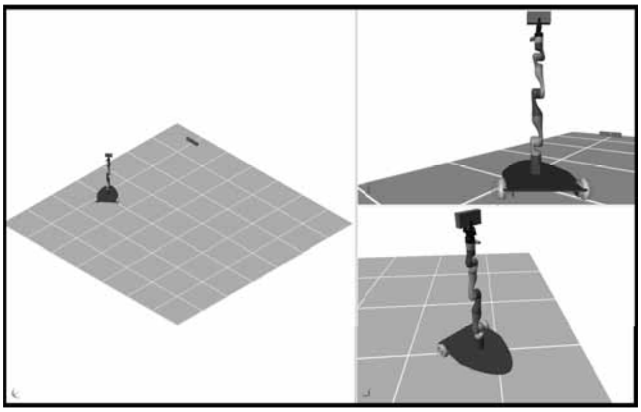


Figure 11: Task scheduler: Drive towards the destination where the bricks should be placed.

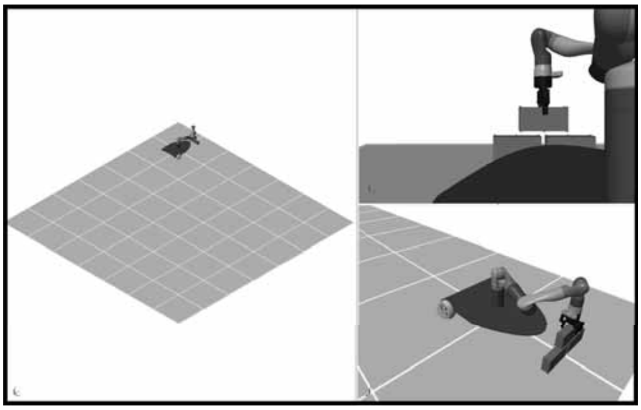


Figure 12: Task scheduler: Place bricks at the destination to construct a wall using the IK solver in Simscape.

Next, the power needed for the articulated gripper arm to pick the third, and last, brick (see Figure 13 accordingly) is investigated.

The power consumption for each joint can be computed using the following equation:

$$P = t \cdot \omega \tag{2}$$

where t and ω stand for the angle by which each wheel of the mobile base rotates and the corresponding angular (rotational) velocity. Simscape offers the *Rotational Power Sensor* block [18], among other sensor blocks, that can be leveraged to compute the power of a rotational mechanical system, see Equation 2. Figure 14 shows the physical conserving ports of this sensor block, which are physical connections that conserve the mechanical system's rotational energy and the output signal port that senses the corresponding rotational power.

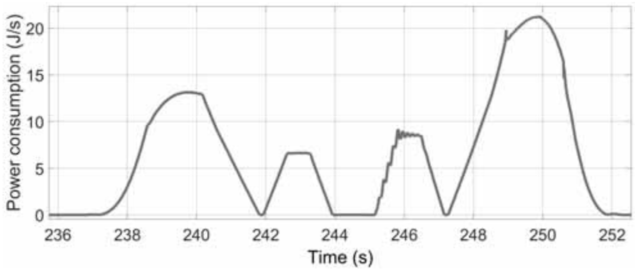


Figure 13: Power consumption of the articulated robot arm when picking up the last brick.

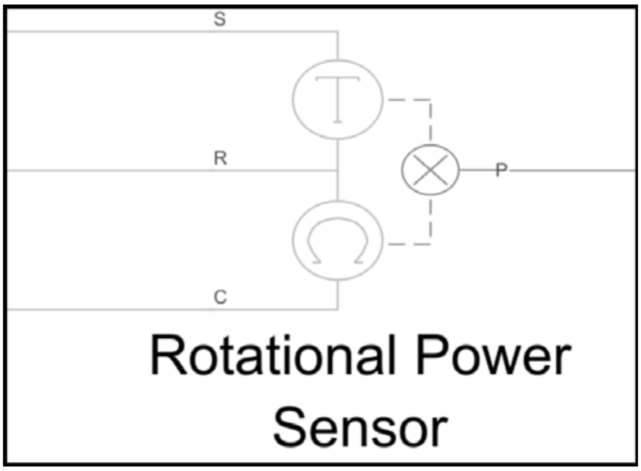


Figure 14: Power consumption of the articulated robot arm when picking up the last brick.

Joints in Simscape Multibody enable coupling different parts of a multibody system by introducing DOFs that may restrict the motions as needed for the corresponding application. Using a Weld Joint between the mobile base of the articulated gripper arm has the desirable effect of mounting the articulated gripper arm onto the mobile base rigidly since the mobile base can freely move on the ground in this case.

The total power consumption needed for the movement of the mobile base is also computed in the same manner. The results are summarized in Figure 15.

The power needed for the mobile base can be exclusively attributed to the actuation of the two rear wheels according to the pure pursuit algorithm for path following and the corresponding *P*-controller. The relatively short instances where the power consumption becomes negative are worth noting. These can be attributed to the moments when the velocity is reduced as the mobile base approaches the destination using the employed triangular velocity profile u sing t he p roportional controller.

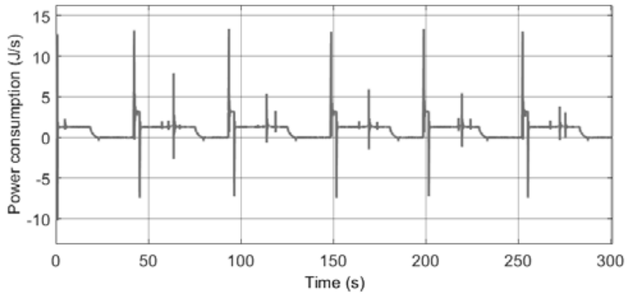


Figure 15: Power consumption of the mobile base throughout the simulation.

When the robot's velocity is reduced by motion input, kinetic energy flows back into the electric motor. These instances can be considered as equivalent to regenerative breaking.

The students were able to ramp up quickly on the underlying topics even though they weren't experts in robotics nor familiar with the computational platforms employed prior to starting the project. They felt comfortable discussing advanced topics in robotics by the end of the project, such as physical modeling, control algorithms, inverse kinematics, and gaits among other. They proposed appropriate solutions to the challenges faced throughout the project.

This project-based learning experience allowed the students to gain significant knowledge in robotics, the employed computational platforms, and team collaboration by working on a project with high practical impact.

4 Conclusions

This work underscores the importance of using adaptable and flexible simulation platforms, such as MATLAB, Simulink, and Simscape, to conduct courses based on Project-Based Learning, which enables students familiarize themselves with prominent industrial methods and workflows.

Moreover, the importance of simulation for the early-stage design of robotic systems in the context of advanced engineering applications is also highlighted herein. It emphasizes the effective and efficient modeling and simulation of robotic multibody systems using Simscape Multibody, Simulink, and MATLAB.

These tools offer the flexibility to choose the level of detail necessary for the required system fidelity.

For example, we demonstrate that while the mobile base motors are modeled with a basic DC electric circuit, the articulated robotic arm's joint motion is simulated directly without specifying the power source. This approach simplifies early-stage design processes by focusing on essential elements. Additionally, we highlight that control design can be directly integrated within Simulink, allowing for easy computation and sensing of the required power using Simscape sensing blocks.

In conclusion, simulation is essential for any stage design of such robotic systems. It provides critical insights into the model under consideration that can be used to enhance productivity, robustness, and eventually, safety when such complex engineering solutions are deployed in practical applications.

Moreover, simulations are widely used to produce data to train reinforcement learning models, create policies for optimal robot control, or train surrogate models from multibody systems via *Deep Learning* to speed up computational time, see for instance in [19].

To this end, modern teaching methods, such as Flipped Classroom, Project-Based Learning, Active Learning, etc., can be considerably enhanced by using industry-mature simulation plat-forms like the ones used in this study.

Publication Remark

This contribution is the improved version of the conference version published in
Tagungsband Langbeiträge ASIM SST 2024,
ARGESIM Report AR 47,
ISBN ebook: 978-3-903347-65-6, pp 95-102,
DOI 10.11128/arep.47.a4722

References

[1] Dimitra Kokotsaki, Victoria Menzies, and Andy Wiggins. Project-based learning: A review of the literature. *Improving schools*, 19(3):267–277, 2016.

[2] José Alberto Naves Cocota, Thiago D'Angelo, and Paulo Marcos de Barros Monteiro. A project-based learning experience in the teaching of robotics. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, 10(4):302–309, 2015.

[3] Ignacio de Los Rios, Adolfo Cazorla, José M Díaz-Puente, and José L Yagüe. Project–based learning in engineering higher education: two decades of teaching competences in real environments. *Procedia-Social and Behavioral Sciences*, 2(2):1368–1378, 2010.

[4] Andreas Apostolatos and Sebastian Gross. Teaching advanced topics in numerical engineering using project-based learning. In *2024 IEEE Global Engineering Education Conference (EDUCON)*, pages 01–10. IEEE, 2024.

[5] Kinova Robotics. Gen3 robotic arm. `https://www.kinovarobotics.com/product/gen3-robots`. Accessed: 2025-03-07.

[6] Steve Miller. Industrial robot models in simscape. GitHub, 2025. Accessed: 2025-03-07.

[7] MathWorks. Online training suite. `https://matlabacademy.mathworks.com`. Accessed: 2025-03-07.

[8] MathWorks. simimport function. `https://www.mathworks.com/help/sm/ref/smimport.html`. Accessed: 2025-03-07.

[9] MathWorks. Kinematicssolver. `https://www.mathworks.com/help/sm/ref/simscape.multibody.kinematicssolver.html`. Accessed: 2025-03-07.

[10] SolidWorks. Solidworks full tutorial – mobile car. YouTube video. `https://www.youtube.com/watch?v=U8WFaWKi3yo`, Accessed: 2025-03-07.

[11] MathWorks. Simscape multibody link. `https://www.mathworks.com/help/smlink/ug/installing-and-linking-simmechanics-link-software.html`. Accessed: 2025-03-07.

[12] MathWorks. Spatial contact force block. `https://www.mathworks.com/help/sm/ref/spatialcontactforce.html`. Accessed: 2025-03-07.

[13] MathWorks. Pure pursuit block. `https://www.mathworks.com/help/robotics/ref/purepursuit.html`. Accessed: 2025-03-07.

[14] MathWorks. Matlab function. https://de.mathworks.com/help/simulink/slref/matlabfunction.html. Accessed: 2025-03-07.

[15] MathWorks. Weld joint. https://www.mathworks.com/help/sm/ref/weldjoint.html. Accessed: 2025-03-07.

[16] MathWorks. Motor & drive (system level). https://www.mathworks.com/help/sps/ref/motordrivesystemlevel.html. Accessed: 2025-03-07.

[17] MathWorks. Rotational multibody interface. https://www.mathworks.com/help/simscape/ref/rotationalmultibodyinterface.html. Accessed: 2025-03-07.

[18] MathWorks. Rotational power sensor. https://www.mathworks.com/help/sdl/ref/rotationalpowersensor.html. Accessed: 2025-03-07.

[19] MathWorks. Twt gmbh develops new workflow for tuning automotive suspension designs using deep learning and multibody simulation. https://de.mathworks.com/company/user_stories/twt-gmbh-develops-new-workflow-for-tuning-automotive-suspension-designs-using-deep-learning-and-multibody-simulation.html. Accessed: 2025-03-07.

The Filtering Effect on Simulated Signals under Consideration of Entropy Methods

Alexander Edthofer*, Andreas Körner

Institute of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria;
*alexander.edthofer@tuwien.ac.at

SNE 35(2), 2025, 93-97, DOI: 10.11128/sne.35.sn.10733
Selected ASIM SST 2024 Postconf. Publication: 2024-12-10
Received Improved: 2025-03-12; Accepted: 2025-03-20
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. In this work, we investigate the impact of lowpass filters on two entropy methods, Permutation Entropy (PE) and Entropy of Difference (EoD), using simulated noise signals. Colored noise, specifically white, pink and brown noise, was generated and filtered with passband frequencies of 30 Hz, 40 Hz and 50 Hz, alongside unfiltered signals. The PE and EoD values were computed to analyze the effects of filtering. The results indicate that both entropy measures decrease with lower passband frequencies. PE can effectively distinguish between pink and brown noise with and without a lowpass filter, while EoD shows similar differentiation only with filtering. These findings highlight the sensitivity of entropy measures to lowpass filtering, with implications for their application in EEG analysis.

Introduction

Simulated signals such as noise are often used for modeling and comparison of situations or occurrences of real-world phenomena.

Pink noise is very common in biosystems, as these are stochastic, self-organizing and their equilibrium is at the lowest energy level possible [1]. In the context of biomedicine, noise can be thought of as an idealized or abstracted signal. The brain activity of awake humans resembles pink noise when measured by an electroencephalogram (EEG) [2].

During unconsciousness, higher frequencies are not as present compared to the awake stage. Therefore, brain activity in this stage can be better compared to brown noise [3].

In the field of EEG analysis, usually there are bandpass filters applied on the measured signals [4].

The permutation entropy (PE) [5], first introduced in 2002, is a commonly used parameter in research in this field of application [6, 7], the entropy of difference (EoD) [8] is new and not yet established in EEG analysis, but seems promising. In this work, we investigate the impact of lowpass filters on these two entropy methods under the use of raw simulated signals using colored noise.

The effect of linear filters on white noise using the PE has already been shown in [9]. However, no other colored noise was considered. In the case of EoD, no such research has been conducted.

The computations were performed on a laptop with 16 GB RAM, an AMD Ryzen 5 5500U processor with operating system Microsoft Windows 10 Pro using MATLAB version R2023b.

1 Methods

This section introduces the methods used to analyze noise signals and their entropy measures.

Firstly, the properties of different noise signals, in particular, white, pink, and brown noise, are described, including their spectral characteristics and their generation by stochastic processes.

Secondly, an explanation of the two entropy-based metrics for analyzing time series follows, i.e. PE, which is based on the frequency distribution of ordinal patterns, and EoD, which considers changes between adjacent values.

These methods allow a quantifiable assessment of the complexity and structure of stochastic processes and are applied to simulated signals in the remainder of this paper.

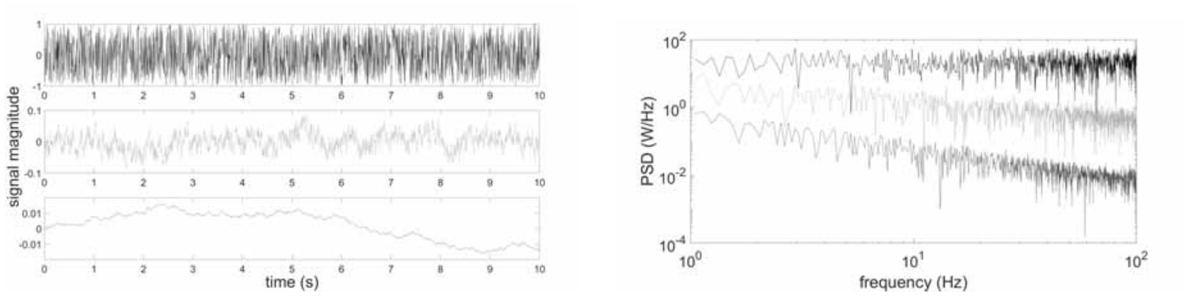


Figure 1: The left picture shows an example for white (top), pink (middle) and brown (bottom) noise signals with white depicted in black and pink and brown in their respective colours. A sampling frequency of 200Hz and a duration of 10s are chosen. The magnitude of each signal is illustrated on the y-axis over the time given in seconds on the x-axis. The corresponding PSDs are given in the right picture. The `loglog` plot shows the equal distribution of the frequencies for the white noise and a decrease in power when the frequencies increase for pink and even more for brown noise.

1.1 Noise signals

Stochastic processes can be used to generate noise signals. The most common ones for model analysis are white, pink and brown noise [10], which also coincide with the simulated signals for the application of EEG analysis.

Their power spectral densities (PSD) in general are given by $S(f) = \frac{L(f)}{|f|^\alpha}$ with L being a positive, slowly varying or even constant function. White noise has a uniform distribution of the frequencies, i.e. $S_w(f) = L(f)$, which means $\alpha = 0$.

A mathematical description is derived by the time-derivative of a Brownian motion process [11]. Pink noise is defined with $\alpha = 1$ as $S_p(f) = \frac{L(f)}{|f|}$, i.e. higher frequencies appear with lower amplitude. Brown noise has an even stronger decrease than pink as $\alpha = 2$, which results in $S_b(f) = \frac{L(f)}{|f|^2}$.

Figure 1 shows the course of white, pink and brown noise with their respective PSDs. The simulated signals in this work are generated using the MATLAB function `dsp.ColoredNoise`.

The noise signals are compared with two entropy methods. There are different lowpass-filters applied to the signals, which are compared as well. For this, the MATLAB function `lowpass` is used. Figure 2 shows the course of the different colored noise and their respective PSDs with a 30Hz lowpass filter.

1.2 Permutation entropy

The PE was first introduced in [5]. A given times series $(x_t) = (x_1, ..., x_N)$ is divided in tuples of length m, which is called the order.

For each tuple, an ordinal pattern is determined, for length m there are $m!$ possible combinations. The PE is defined as

$$PE = -\frac{1}{\log(m!)} \sum_{i=1}^{m!} p_i \log p_i, \qquad (1)$$

where p_i defines the probability of occurrence of pattern i and the base of the logarithm is two.

The coefficient $-\frac{1}{\log(m!)}$ represents a normalization factor, such that $PE \in [0,1]$. If two values in one tuple are equal, i.e. $x_i = x_j$, we choose the rule that $x_i < x_j$ for $i < j$. A detailed mathematical description is given in [7, 12].

1.3 Entropy of difference

An alteration of the PE is the EoD, defined by Pasquale Nardone in [8]. Again, a time series (x_t) is divided in the same amount of tuples of length m.

For this method, one considers the neighboring values within a tuple and compares them. The encoding is given only by $+$ and $-$, depending if there is an increase or decrease between the values. If two neighboring values would be equal, we again apply the rule that $x_i < x_j$ for $i < j$. For tuples of length m, there are 2^{m-1} possible patterns that can be achieved. The EoD is then defined as

$$EoD = -\frac{1}{m-1} \sum_{i=1}^{2^{m-1}} p_i \log p_i, \qquad (2)$$

where p_i again defines the probability of occurrence of pattern i.

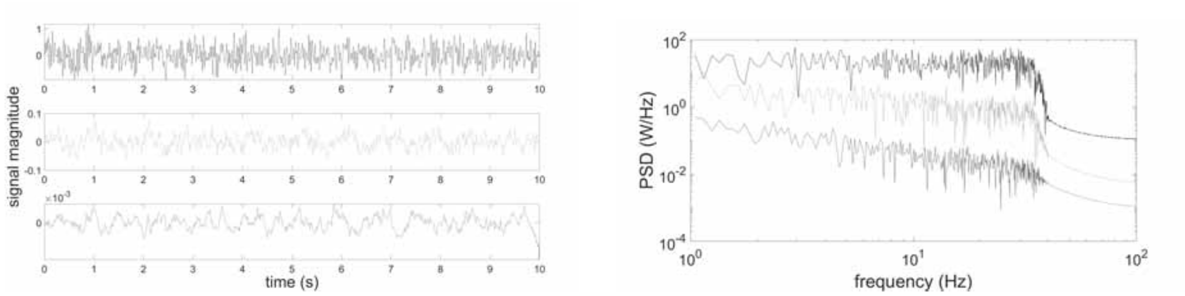


Figure 2: The left picture shows an example for white (top), pink (middle) and brown (bottom) noise signals with a 30Hz lowpass filter with white depicted in black and pink and brown in their respective colours. A sampling frequency of 200Hz and a duration of 10s are chosen. The magnitude of each signal is illustrated on the y-axis over the time given in seconds on the x-axis. The corresponding PSDs are given in the right picture. In comparison to the unfiltered case, the `loglog` plot shows a sharp decline at the lowpass-frequency.

The base of the logarithm is two, such that the coefficient $-\frac{1}{m-1}$ is also a normalization factor, i.e. $EoD \in [0,1]$. A detailed description of the EoD is given in the publication [12].

2 Results

For our study, colored noise with $5 \cdot 10^6$ sample points was created.

Firstly, the simulated signal is lowpass filtered by a passband frequency of either 30 Hz, 40 Hz or 50 Hz. Considering also unfiltered signals, this makes a total of four different scenarios. These passband frequencies were chosen as these are also used in the application of EEG analysis [4].

Secondly, the sample was rearranged to 1665 vectors each containing 3000 values.

Comparing such a vector to a recorded signal with a sampling frequency of 200 Hz, this would correspond to 15 s. As stated by [4], this is indeed a reasonable setting in terms of sampling frequency and signal duration in the application field of EEG analysis.

Next, the simulated signals were decoded in the respective patterns of the PE and EoD and afterwards the corresponding entropy values were calculated.

The results of the two different entropies are given in the case of orders $m = 3$ and $m = 7$. For the PE, the most common orders are between 3 and 7 [5].

A graphical representation of the results for the 1665 samples is given in Figure 3 for $m = 3$ using boxplots, which were created using the MATLAB function `boxplotgroup`.

The y-axis refers to the entropy values, which are defined in equations (1) and (2), shown between 0.5 and 1 as these are the minimal and maximal value that appear in our study. The four different filter scenarios are indicated on the x-axis.

For order 3, a decrease of the PE value from white to pink to brown noise is observable, independent of the applied filter. EoD does not show the same behavior, as for no lowpass filter, there is an increase of the values from white to brown noise.

In general, a decrease in the lower passband frequencies is observable for PE and EoD with a more prominent decrease in the case of PE.

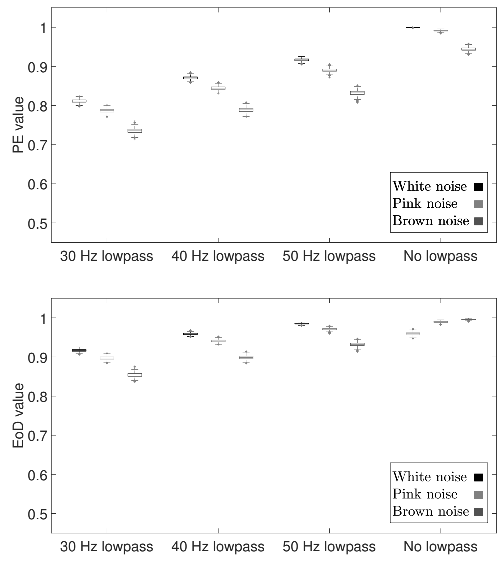


Figure 3: Entropy values of order $m = 3$ and different filters for white, pink and brown noise.

This is reasonable because, when lowpass filtering is done, fewer patterns occur. The PE can distinguish between pink and brown noise for all four scenarios as none of the respective boxplots overlap. The EoD manages this task only for the three scenarios in which lowpass filtering was applied. For the PE with $m = 3$ and no filtering, the results coincide with the ones of [10] and therefore confirm them.

The results for order $m = 7$ are shown in Figure 4, created with the same function and settings as before. The PE for order 7 shows a similar course to $m = 3$ although the decrease for a lower passband frequency is even stronger. However, it still separates pink and brown noise well, as the boxplots again are not overlapping.

The EoD for order $m = 7$ cannot distinguish between pink and brown noise even for the filtering scenarios. The values for white noise are in all cases lower than those for pink noise.

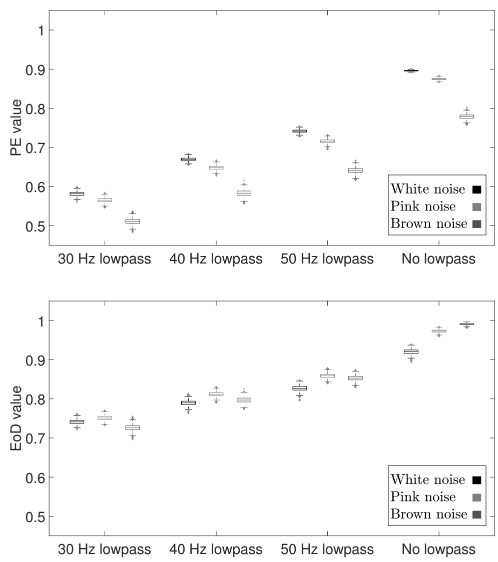


Figure 4: Entropy values of order $m = 7$ and different filters for white, pink and brown noise.

3 Discussion

In this work, we tested different lowpass filters on colored noise and investigated their impact on PE and EoD.

Passband frequencies of 30 Hz, 40 Hz and 50 Hz were compared with no lowpass filtering, as these three are most commonly used in the field of EEG analysis [4].

The results show that the EoD behaves differently, if no filter is set. For any other of the three scenarios, EoD behaves similar to the PE for order $m = 3$. For the order $m = 7$, the results differ, as white noise does not achieve the highest EoD value, but pink noise.

One can see as well that the lower the passband frequency was set for the lowpass filter, the lower the entropy values get. For the PE and white noise, this was already indicated in [9].

We also showed the effect on other types of noise as well as a similar impact on the EoD. However, the decrease in value in the latter case is not as strong as for the PE.

We also considered a highpass filter, usually there is a passband frequency of 0.5 Hz, but this did not show any effect on the values of the two entropy methods.

The quality of EoD in comparison to PE, especially in the application of EEG analysis, is done in [12]. Here, we showed that EoD has lower computational cost than PE with an equally good performance of classifying sleep states and even better classification results for vigilance states during anesthesia.

Acknowledgement

The authors would like to thank their research partner the AG Neuromonitoring of Dr. Matthias Kreuzer from the Technical University of Munich for their continued guidance in the field of EEG analysis and, additionally, Clara Horvath for proofreading.

Publication Remark

This contribution is the improved version of the conference version published in
Tagungsband Kurzbeiträge ASIM SST 2024,
ARGESIM Report AR 46,
ISBN ebook: 978-3-903347-64-9,
Volume DOI 10.11128/arep.46, pp 29-31.

References

[1] Szendro P, Vincze G, Szasz A. Pink-noise behaviour of biosystems. *European Biophysics Journal*. 2001; 30:227–231.

[2] Buzsaki G. *Rhythms of the Brain*. Oxford university press. 2006.

[3] Colombo MA, Napolitani M, Boly M, Gosseries O, Casarotto S, Rosanova M, Brichant JF, Boveroux P, Rex S, Laureys S, et al. The spectral exponent of the resting EEG indexes the presence of consciousness during unresponsiveness induced by propofol, xenon, and ketamine. *NeuroImage*. 2019;189:631–644.

[4] Sanei S, Chambers JA. *EEG signal processing*. John Wiley & Sons. 2013.

[5] Bandt C, Pompe B. Permutation entropy: a natural complexity measure for time series. *Physical review letters*. 2002;88(17):174102.

[6] Berger S, Schneider G, Kochs EF, Jordan D. Permutation entropy: too complex a measure for EEG time series? *Entropy*. 2017;19(12):692.

[7] Edthofer A, Feldhammer I, Fenzl T, Körner A, Kreuzer M. Permutation entropy as a conceptual model to analyse brain activity in sleep. In: *EUROSIM Congress*. Springer. 2023; pp. 205–218.

[8] Nardone P, Sonnino G. Entropy of Difference: A New Tool for Measuring Complexity. *Axioms*. 2024; 13(2):130.

[9] Dávalos A, Jabloun M, Ravier P, Buttelli O. The impact of linear filter preprocessing in the interpretation of permutation entropy. *Entropy*. 2021;23(7):787.

[10] Chen Z, Ma X, Fu J, Li Y. Ensemble Improved Permutation Entropy: A New Approach for Time Series Analysis. *Entropy*. 2023;25(8):1175.

[11] Kasdin NJ. Discrete simulation of colored noise and stochastic processes and 1/f/sup/spl alpha//power law noise generation. *Proceedings of the IEEE*. 1995; 83(5):802–827.

[12] Edthofer A, Ettel D, Schneider G, Körner A, Kreuzer M. Entropy of difference works similarly to permutation entropy for the assessment of anesthesia and sleep EEG despite the lower computational effort. *Journal of Clinical Monitoring and Computing*. 2024;pp. 1–14.

Concept Development for Coupling Simulation and Machine Learning in Supply Chains

Yen Mai Thi[1], Marc Münnich[2], Florian Zumpe[2], Mario Callefi[3], Pierre Grzona[3]*

[1]Westsächsische Hochschule Zwickau, Kornmarkt 1, 08056 Zwickau, Germany

[2]Fraunhofer-Institut für Werkzeugmaschinen und Umformtechnik IWU, Reichenhainer Straße 88, 09126 Chemnitz, Germany

[3]Technische Universität Chemnitz, Erfenschlager Str. 73, 09125 Chemnitz, Germany
*pierre.grzona@mb.tu-chemnitz.de

SNE 35(2), 2025, 99-103, DOI: 10.11128/sne.35.sn.10734
Selected ASIM SST 2024 Postconf. Publication: 2024-12-10
Received Revised: 2025-03-25; Accepted: 2025-04-04
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. This study provides insights into the development of a combined simulation and machine learning approach in the field of additive manufacturing (AM).
The systems engineering (SE) methodology is used to determine the relationship between different engineering objectives in the area of semantic supply chain modelling to validate the results of a matchmaking model for the digital AM supply platform to increase its network resilience. The concept is used as a basis for further implementation.

Introduction

The aim of this study is to present the current state of research on the coupling of simulation and machine learning (ML) in the supply chain environment as part of a research project to create a platform-based simulation service.
During the last year, several prerequisites have been elaborated, one of which is the possible coupling scenarios for simulation and ML [1]. From four cases based on VDI 3633 Part 12:2020, the combination of simulation followed by a ML approach was selected as the basis for further investigation. [2].

In addition, the different approaches to determining the resilience of a supply chain have been examined by Grzona et al. [3].

Based on the literature review, different use cases were identified and the targets for the simulation studies. Time and cost were the most mentioned objectives for investigation in the literature and the use of reinforcement learning and neural networks as ML's techniques of ML. The survey also identified the possibility of increasing the ML knowledge of simulation experts.

1 Methodology

The research objective to be achieved is to create a concept for the coupling of simulation and ML in an exemplary value network to assess the previously mentioned objectives.

This leads to two Research Questions (RQ):

- RQ1: What can an exemplaric value network look like?
- RQ2: How can a coupling concept look like?

An SE approach was chosen to further develop the concept for the implementation of the coupling case. The SE approach is a well-established methodology for designing systems that incorporates the principles of systems thinking.

The SE process model and the methods and tools used in the problem-solving process support a systematic transfer from an actual state to a desired state.

Understanding the elements and relationships within the system and their relationships to the environment and environmental systems is a fundamental aspect of systems thinking. [4]

These systematic perspectives can also be used in conjunction with technical systems in socio-technical systems, which incorporate human and societal aspects [5]. In addition, the entire lifecycle of a system is considered, from its development to its realisation and use [4].

2 Concept Development

Ahrens [5] proposed a six phase approach for an SE cycle (Figure 1). As the scope of the design is determined by the project, a brief introduction will be given, followed by an explanation of the connection to the matchmaking process. This is followed by a description of the technical objectives that lead to the methods used to produce the overall concept in the penultimate step.

2.1 Reasons for the design project

The aim of the project is to develop a digital platform for AM to increase the resilience of value networks using artificial intelligence and ontologies.

Since the data of the system elements - in this case AM resources - are based on the semantic data model, it is necessary to integrate these concepts into the overall process [6].

The AM process chain has five phases, each with its own tools, models and methodological requirements. Product design aims to create a digital object using Computer Aided Design (CAD) software.

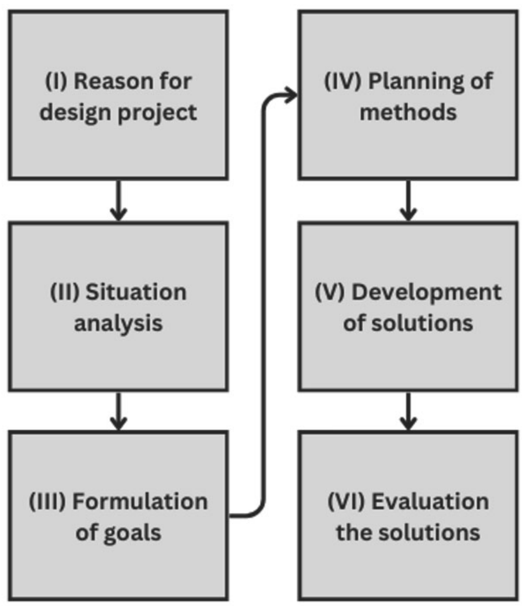


Figure 1: Six phases of SE process according to Ahrens [5].

The pre-process is typically done in Computer Aided Planning (CAP) software to optimise the build space, slice the model and set the machine parameters.

During the printing process, several measures are taken to monitor the build process and, if necessary, adjust the machine parameters on the fly to reduce the error rate. This is supported by Computer Aided Manufacturing (CAM) software. After printing, post-processing typically involves a number of steps, including cleaning and separation of the support structures and verification of the physical product and process dimensions.

The final stage is the finishing process, which uses a combination of techniques, including thermal finishing to harden the products, mechanical methods and quality control, which may be destructive or non-destructive. Each of these steps takes a certain amount of time [7].

2.2 Situation analysis

The Simulation/ML core should be used to validate and support the general matchmaking process on the platform [8]. This is driven by the semantic data model for the capabilities of the different roles in the process chains.

Out of the process chain, a graph is generated for the value network, which requires multiple combinations of suppliers and partners to be found using a matchmaking service.

At this stage, the behaviour of the network is not evaluated or assessed in terms of resilience, time, price, or quality. These are then used to develop the models for the Simulation/ML.

2.3 Formulation of goals

The technical goals were elaborated in a continuous group work process. As previously mentioned, a semantic database is required. For this project, a specific AM logistics ontology was employed, which is connected to the AM ontology [6].

As the concept needed to be proven first, the focus was on incorporating existing simulation tools rather than developing new ones. Therefore, Siemens Tecnomatix Plant Simulation, AnyLogic, and Anylogistix were chosen for their ability to interact with external software through interfaces.

In addition to working with these tools, the solution should be capable of analysing data from multiple simulation experiments. This will provide a data basis for determining whether a reinforcement learning or neural network approach can be used.

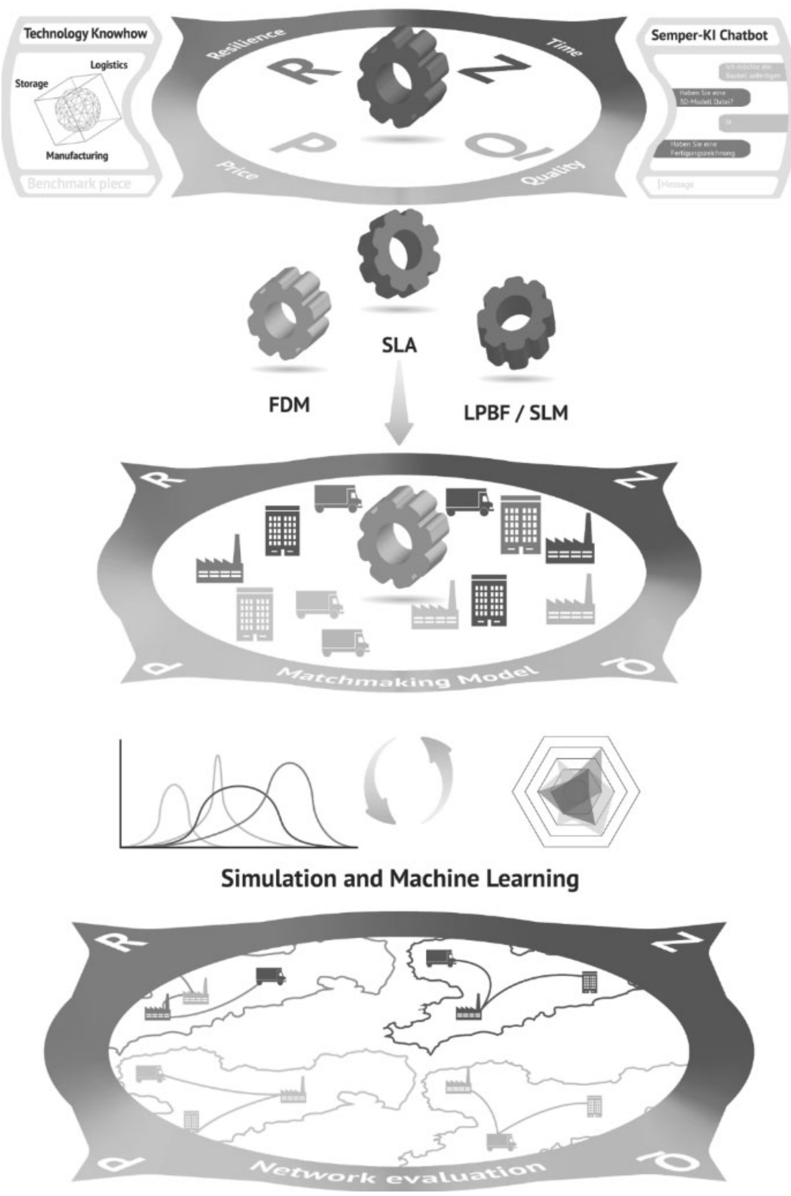


Figure 2: Meta model of matchmaking model and Simulation/ML core © InfAI e.V. Leipzig.

Due to the involvement of stakeholders from various domains with differing levels of expertise, the development of a highly comprehensive model is essential. Initially, the model should address time and cost, with the flexibility to adapt to additional aspects as required.

2.4 Planning of methods

For the problem-solving cycle, the case study method was selected because of the exploratory and unique nature of the problem to be solved. [9].

The first test case was defined based on a product consisting of three different parts, manufactured by the three partners and delivered to one customer. It is important to note that the partners have different production capabilities, and that the product requires a number of steps of post-processing. These steps consist of refining the parts, assembly, quality control, packaging, and delivery to the customer.

2.5 Development of the solution

The concept pipeline has been developed (Figure 3) to meet the previously outlined requirements. To connect the different systems, KNIME, an open-source and versatile data mining tool, is used. In parallel, it allows for the use of ML techniques and fulfils the node-oriented interface design goals.

In order to provide a precise answer regarding the lead time, it is necessary to consider the specific capabilities of our partners, their geographical location, and the processing times.

This can be evaluated using Anylogistix in the experiments. The various AM technologies have a considerable impact.

For instance, the powder bed fusion process has different optimisation criteria compared to a fused deposition modelling process in terms of the use of building space and the number of parts produced in parallel [10].

With regard to the cost assumption, it was necessary to evaluate not only the time required for printing, but also to consider the various resource costs in the network and the relationships between them [11]. This resulted in a partner and activity-based cost formula for each subject in the value network. The solution was developed by one currently self-defined case, which follows the assumption that each partner could have multiple roles in the supply chain.

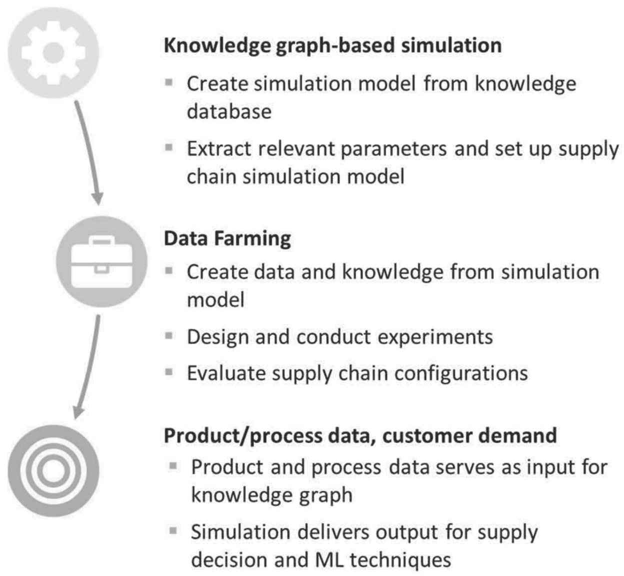


Figure 3: Pipeline for Simulation/ML ©Fraunhofer IWU, Chemnitz.

This role model can be understood on the ontological level as domain-specific knowledge. Additionally, the knowledge for the specific value network is represented by the knowledge graph derived from the database and used by the simulation software.

The results are then transferred back to the knowledge graph on the platform. This data will then be used to train the ML algorithms.

2.6 Evaluation

Due to the inherent characteristics of the SE process, it was not possible to fully evaluate the results at the system level; instead, the evaluation was carried out at the level of the system components.

The simulation will be run using both simulation tools connected to the KNIME data flow model, and their results will be compared to a deterministic model of the value network. The results of the deterministic model will then be recalculated with classic spreadsheet software to validate the gained solutions.

3 Conclusion and Limitations

The research questions have been addressed, and the concept has been proven in part with only one case study, which only includes manufactured parts.

Further evaluation is required with different and more complex value networks. The study provides an overview of a data-driven platform's overall data flow process involving simulation techniques.

Additionally, the use of slicing software such as Slic3r or hardware-specific tools is recommended to more accurately estimate the in-process time. As the data backend is not yet fully available, the data stream as input was manually added, so the amount of data is not yet sufficient to run ML techniques in a meaningful way. This necessitates further incorporation of the knowledge graph stack in a proof-of-concept stage.

This will also lead to further investigations to assess which type of ML technique is more suitable for the connection with the simulation tools used. Due to the project's characteristics, no difference or performance comparison with other approaches is made, which would provide more insight into possible performance or quality advantages at the decision level.

Acknowledgments

The project on which this report is based was funded by the German Federal Ministry of Economic Affairs and Climate Protection under the funding code MM - MMMKO01416521 - 0I1MK22001G. The responsibility for the content of this publication lies with the authors.

Disclosure of Interests.

The authors have no competing interests to declare that are relevant to the content of this article.

Publication Remark

This contribution is the improved version of the conference version published in

Tagungsband Kurzbeiträge ASIM SST 2024,
ARGESIM Report AR 46,
ISBN ebook: 978-3-903347-64-9,
Volume DOI 10.11128/arep.46, pp 25-28.

References

[1] Münnich M, Zumpe F, Grzona P. Simulation und Maschinelles Lernen in Supply Chains. In Krull C, Commerell W, Durak U, Körner A, Pawletta T, editors. ASIM WS 2023 Kurzbeiträge und Abstract-Beiträge. *ASIM STS, GMMS & EDU Workshop*; 2023; Magdeburg Germany. Vienna: ARGESIM. pp. 15-18. doi: 10.11128/arep.22.

[2] VDI Association of German Engineers e.V.: Simulation of systems in materials handling, logistics and production: Simulation and optimisation (VDI 3633 Part 12). Berlin: Beuth Verlag GmbH, 2020.

[3] Grzona P, Mai TY, Zumpe F, et al. Requirements for a ML- and platform-based simulation service. In Bergmann S, Feldkamp N, Souren R, Straßburger S, editors. 20. ASIM Fachtagung Simulation in Produktion und Logistik 2023. *20. ASIM Fachtagung Simulation in Produktion und Logistik*; 2023; Ilmenau Germany. Ilmenau: Universitätsverlag Ilmenau. pp. 187-196. doi: 10.22032/dbt.57805

[4] Haberfellner R, Weck O de, Fricke E, et al. *Systems Engineering*. Cham: Springer International Publishing; 2019. 458 p.

[5] Ahrens V. *Systems Engineering and Management*. Wiesbaden: Springer Vieweg; 2024. 612 p.

[6] Hartmann H. Semper-KI Ontology V.09. Accessed: Jun 2, 2024 [Online]. Available: https://schema.semper-ki.org/

[7] Lachmayer R, Ehlers T, Lippert RB. *Design for Additive Manufacturing*. Berlin: Springer Berlin Heidelberg; 2024. 257 p.

[8] Chen X, Thi YM, Sprigode J, et al. Partner Selection in Additive Manufacturing Networks. In Thürer M, Riedel R, von Cieminski G, Romero D, editors. Advances in Production Management Systems. Production Management Systems for Volatile, Uncertain, Complex, and Ambiguous Environments. *43rd IFIP WG 5.7 International Conference, APMS 2024*; 2024 Sep; Chemnitz Germany. Chem: Springer International Publishing. pp. 424–438. doi: https://doi.org/10.1007/978-3-031-71629-4_29

[9] Yin RK. *Case study research and applications*. 6th Edition. Los Angeles, California: SAGE Publications; 2018. 352 p.

[10] Tagliaferri V, Trovalusci F, Guarino S, et al. Environmental and Economic Analysis of FDM, SLS and MJF Additive Manufacturing Technologies. *Materials*. 2019; 12(24). doi: 10.3390/ma12244161.

[11] VDI Association of German Engineers e.V.: Activity-based cost analysis in internal logistics (VDI 4405 Part 1). Berlin: Beuth Verlag GmbH, 2003.

Process Model for Integration of Speech Recognition and Understanding in Multiple Remote Tower Control Simulations

Oliver Ohneiser[1,2*], Hartmut Helmke[1], Sebastian Schier-Morgenthal[1], Umair Ahmed[2]

[1]Institute of Flight Guidance, Department Controller Assistance, German Aerospace Center (DLR), Lilienthalplatz 7, 38108 Braunschweig, Germany; *oliver.ohneiser@dlr.de, https://orcid.org/0000-0002-5411-691X

[2]Institute for Informatics, Clausthal University of Technology, 38678 Clausthal-Zellerfeld, Germany

SNE 35(2), 2025, 105-115, DOI: 10.11128/sne.35.tn.10735
Submitted: 2025-05-20; Revised: 2025-05-29
Accepted: 2025-06-10
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Simulations – especially human-in-the-loop real-time simulations – are important in the air traffic control (ATC) domain to train controllers and to test new features for controller working positions. One important reason for such simulations is the measurement of human workload. Verbal communication of aviation operators – contributing to this workload – is a central mean for safety and efficiency of air traffic. Speech recognition and understanding (ASRU) has reached pre-industry level, is about to enter operations, and therefore will become a vital part in training. The technology affects working procedures and reduces controller workload by roughly 20%. Thus, ASRU must be considered in simulations.

This paper describes a process model to integrate ASRU in ATC simulations. The model consists of three steps for efficient integration and adaption of ASRU: (1) collection of in-domain speech data for tuning of acoustic and language models, (2) compilation of configuration files and adaptation of speech understanding algorithms, and (3) manual checking of automatic transcriptions and extracted, semantic meanings of speech utterances.

We evaluate the process using a multiple remote tower environment case study. In this study, recognition error rates for words and callsigns were reduced by a factor of three compared to first simulations and command recognition rates increased from 81% to 92%.

Similarly feasible results are expected for other new ATC simulations with ASRU using the proposed process model.

Introduction

Simulating air traffic control (ATC) is crucial for training of air traffic controllers (ATCos) outside of their operational environment. They can train new or seldom executed operational procedures and test new features for controller working positions before potential deployment. Radio telephony communication between ATCos and cockpit crews is in general a crucial part of these simulations [1].

The transformation of operational, analogue voice signals into spoken words, intended meanings, and further provision of the digitalized ATC commands for downstream applications has been subject to numerous research projects [2]. ASRU is evaluated ready for operational usage and as such is expected to become a standard feature for future training and system development. As ASRU impacts ATC procedures in a way that workload of ATCos is reduced by 20% (cf. [3]) it must become a standard feature of ATC simulators.

Moreover, simulators will benefit from additional available data: Participating simulation staff such as supervisors or simulation pilots can be supported by receiving given ATC commands in real-time, e.g., for assistance, automation, data recording, and analysis.

The major challenge to achieve these benefits is to meet the requirements of the dynamical changing simulation environments including, e.g., airspace characteristics. To cope with these circumstances, we propose an iterative integration approach into ATC simulators with continuous improvement of ASRU.

This approach contains three steps which are repeated multiple times to iteratively improve the steps' quality:

1. Acoustic modelling: Integrating new words/accents,
2. Semantic modelling: Integrating new ATC concepts/commands,
3. Verification: Evaluation of transcriptions and recognized ATC concepts/commands.

We demonstrate the feasibility of the approach using a case study in which a simulation environment with multiple remote tower control is enhanced by ASRU [4].

To achieve reasonable speech recognition and understanding rates, it is important to have a large set of audio recordings from the aviation domain, which can be used as training data. Earlier ASRU projects in ATC used around 30 hours of in-house training data [5], [6].

However, the available open-access corpora are still limited in size compared to other domains as they are specially protected by telecommunication laws. In our context the main issue is, however, that voice recordings for the new application do not exist at all as the simulation training is required before the operational introduction. Furthermore, the important ASRU step of extracting relevant semantic ATC concepts from the transcribed word sequences, predominantly covered the ATC approach and en route environment with operational and simulation audio recordings. The aerodrome environment including tower, multiple remote tower, and apron has only been tackled to a lesser extent in simulated environments.

To cope with these challenges, we further present two different tools with tested user interfaces to (i) record ATC speech data in a structured way and (ii) for automatic transcription of aviation operator utterances. The collected data fed the training data pool of an ASRU module for a close-to-reality ATC multiple remote tower human-in-the-loop (HITL) simulation.

This paper outlines related work in Section 1. Section 2 presents the user interfaces and configuration files to support data recording, transcription, and annotation as required in the three repetitive steps of our process model. A transcription contains the word-by-word utterance content, whereas we use the term annotation for speech understanding, i.e., performing semantic interpretation of word sequences from the transcriptions. Section 3 explains the human-in-the-loop simulation setup for a multiple remote tower simulation, the integration of ASRU, and results on the ASRU performance at different stages. This is followed by conclusions in Section 4.

1 Related Work

1.1 ATC Speech Recognition & Understanding

Communication between ATCos and pilots with mutual understanding is a cornerstone of safe and efficient air traffic [7]. Speech recognition delivers the spoken word sequences of ATC utterances [8]. Instruction understanding extracts the semantic meaning of such word sequences [9]. A combined ASRU module can enable downstream applications or help to assess communication quality parameters [10], e.g., in HITL-simulations or operational environments.

Such an ASRU module for ATC communication follows a series of steps:

- First, aircraft callsigns and ATC commands that will most likely appear in the next ATCo utterances are predicted based on contextual data such as surveillance data [11].
- Second, a speech-to-text engine delivers the recognized sequences of words from analyzed ATC utterances [12].
- Third, a text-to-ATC-concept component extracts the relevant callsigns and ATC commands from the recognized word sequences [13].

These extracted ATC concepts follow a European-wide agreed ontology for semantic annotation of ATC utterances [14]. The performance of the speech-to-text step is measured via word error rates, but more importantly the text-to-ATC-concept step is analyzed via recognition rates and error rates for the extracted ATC concepts [15]. The output of the ASRU module is then used to support ATCos within their workstation displays [4]. The possibility to quickly add new features from ASRU module outputs into displays used in HITL-simulations helps to get swift feedback from human operators on the features' feasibility [16].

An early assessment has been done for the feasibility of ATC automation within a simulation environment [17]. Virtual simulation pilots using an ASRU module can support automating ATC simulations [18].

1.2 Simulators in the Aviation Domain

Simulations are a central mean in ATC [19]. Simple experiments such as effects of modified airspace [20] or landing clearances [21] can be evaluated without human operators.

However, there are many other ATC operation experiments that require human involvement in situation assessment, decisions, and communication. The analysis of such relationships usually requires a HITL-real-time simulation (RTS) [22]. This methodology is as well common for simulations with pilots [23]. HITL-RTS can be scripted with the help of, for example, ATC scenarios to include relevant air traffic situations in the different domains tower, approach, and en route [24].

There exist some simulators at national air navigation service providers (ANSPs) or research institutes. There are even publicly available ATC simulators [25] and open source simulators based on open data [26]. ATC simulators represent very different fidelities regarding their realism and completeness [19]. They usually also focus on just one of the domains, i.e., approach/en route [27] or tower [28]. We focus on the tower environment also comprising remote and multiple remote towers [29]. While speech recognition is available for some simulators to automate or support simulation pilots, speech understanding to enable operational support (e.g., flight strip handling) exists to a much lesser automation degree.

1.3 Speech Recognition & Understanding and its Influence on Controller Workload

The ATCo workload in a HITL-RTS with and without ASRU was objectively measured with the time needed for a secondary task [12].

In the baseline condition, ATCos needed to maintain aircraft radar labels completely manual, i.e., enter the command content via mouse and drop-down menus.

In the solution condition, ATCos were automatically supported by an ASRU system [12].

After compensating sequence effects – depending on the first simulation run was baseline or solution – and eliminating outlier the average time to solve the secondary task was almost six minutes for the baseline condition (347 s), but less than five minutes for the solution condition (289 s).

Hence, the ATCos needed 20% more time to perform a secondary task if they were not supported by automatic radar label maintenance with the ASRU output in their primary ATC task. Thus, ATC simulators without ASRU can cause workload measurement errors of up to 20% in the future compared to operational environments with ASRU.

1.4 Creating Speech Recognition Models

Thanks to Siri or Alexa speech recognition has now reached the general public. However, these engines are not usable for ATC applications due to insufficient recognition performance and data privacy issues.

Recently, some general engines, so called open source end-to-end models like, Whisper [30] or wav2vec [31] gained more and more attention like with an application for ATC [32]. These end-to-end models often come with easier implementation and adaptation processes. This enables also non-speech recognition experts to reach suitable performances in different target areas. These engines have seen already ten thousand of hours of normal English conversation.

Nevertheless, some fine-tuning with, e.g., ten hours of airport dependent data is necessary for ATC applications. This fine-tuning has shown to reduce the word error rate (WER) from 90% to 5% for CoquiSTT [33]. The same was reported for DeepSpeech [34]. The STARFiSH [5] and HAAWAII project [6] used a basic engine, which has already seen a lot of ATC training data. This engine was fine-tuned with roughly 30 hours of domain dependent data. The HAAWAII project started with just one hour of domain dependent data from the Icelandic airspace. This one hour already reduced the WER from 50% to 33% while three further hours reduced it to 20%. This eases the effortful manual transcription task. The ATCO2 project utilizes unsupervised learning on 5000 hours of ATC voice recordings together with context information from radar data [35].

MITRE presents the FAA system DRAAS (DALR Remote Audio Access System), which, in principle, provides access to audio from 129 National Airspace facilities. More than 200,000 hours of silence-reduced audio are recorded each month, i.e., 2-3 billion ATC transmissions per year are recorded. This enables at least unsupervised learning [36].

Currently, end-to-end speech recognition models own the highest potential for the application in HITL-simulations. On one hand they are already trained on a large dataset of formal English language. On the other hand, they can easily be adapted without specific speech recognition expertise. Nevertheless, ATC applications require fine-tuning.

2 Simulation Setups and Tools

2.1 Online ATC Speech Recorder

For the adaptation of ASRU models within ATC applications such as acoustic model, language model, or command extraction model, training data is required to achieve an acceptable performance. Static or quasi-static data that frequently appear in ATC communication such as frequencies and airline names are quite easy to acquire as verbalized speech.

However, depending on the ASRU use case, the dynamic data comprises speech of ATCos and pilots, surveillance data, flight plans, meteorological data, ATC sector configurations, and many more.

The best recording environment for speech (and surveillance) data is the environment, in which the later ASRU-related ATC application is executed. Unfortunately, the targeted environment is not available in all cases as for instance conversion training and validation projects aim on setups which are not established in real operations. Moreover, ATCos' duty time is a rare resource [37] and should be used as little as possible for the preparation of the simulation setup.

Hence, the minimum setup should be easy to use and accessible from remote to provide the required training data for future working procedures without the need for ATCos to travel or train in advance. These requirements can be fulfilled with a website that ATCos can login to. A server as backend could offer a simplified traffic simulation and speech recording option.

One of the biggest challenges of a qualified data set given alternative data recording options is the feasibility for the later ASRU-related ATC application, i.e., the generated data content should be close to the data content expected in the final simulations. Hence, a good option would be a complete remote simulation, i.e., the ATCo manages interactive air traffic like at a normal controller working position with uttered ATC commands that are recorded. Again, a prioritization of requirements should be made in order to deliver a reasonable solution in a reasonable amount of time.

The simplest recording option consists of a sheet of paper with written ATC utterances that ATCos should read while being recorded on a headset. The recordings could help to learn acoustic models, e.g., the sound of ATC domain prosody. However, the language (sequence of words) would already be predefined and not realistically help to learn a language model, because ATCos more or less deviate from the International Civil Aviation Organization (ICAO) phraseology. This needs to be included into the ASRU models.

An enhanced setup version should give ATCos a greater level of freedom to formulate their own utterances just with some basic hints about the air traffic situation. The ATCos would see static figures with air traffic situations, the last utterances of the involved aircraft pilots, and some options on how to react in the current situation in a very basic style with ATC command type suggestions. This forces the ATCos to actively think about the situation, to use predefined aircraft callsigns, runways, airport names, and waypoint names as used in final simulations, and produces a more natural speech comparable to a minimum simulation. Such recorded speech data would support to learn a language model, i.e., the words and word sequences that ATCo utterances operationally contain and a command extraction model, i.e., what the ATCos mean by their utilized phraseology.

Figure **1** shows our implemented prototype of an online ATC speech recorder. As first step after login, the ATCos need to confirm their participation and data upload. One is then asked to walk through 20 different ATC scenarios. The online ATC speech recording environment offers static air traffic situations in a simple implementable map view as shown in Figure 1 and communication info that can be understood when going through the online tutorial with explanation boxes.

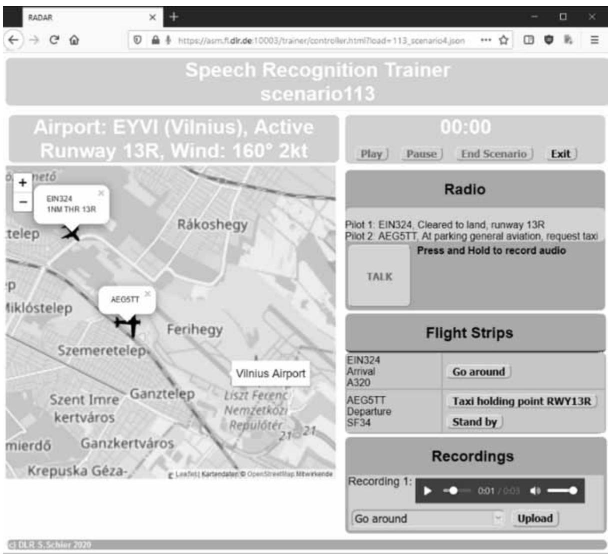


Figure 1: Screenshot of the Online-ATC-Speech-Recorder with a map view (left) and the communication info (right) for a given situation at Vilnius airport with a voice recording.

The map view presents one involved airport with some aircraft information on the left side. The right side shows scenario information (see Figure 1, light blue with white font), last radio calls (see Figure 1, light grey; from Pilot 1 and Pilot 2), flight strips of both involved aircraft with reasonable commands in the given situation (see Figure 1, yellow and blue), and options for recording, labelling, and upload of audio files (see Figure 1, red).

The corresponding ATCo voice utterance is recorded while pushing the "TALK"-button via the ATCo's headset. This speech recording setup forms the second lowest simulator fidelity category 'B' with the used electronic equipment attributed to category 'C' of table 1 in [19].

Nine ATCos from the Lithuanian ANSP and five ATCos from the Austrian ANSP contributed to an online-recording resulting in 667 audio files with one hour of net speech, i.e., each utterance lasts five seconds on average.

2.2 Online ATC Speech Recognition

The recording of audio files can as well be directly connected to an online speech-to-text engine to immediately receive the transcripts of ATCo or pilot utterances in the desired format. The developed browser-based application as shown in Figure 2 utilizes hypertext markup language (HTML), cascading style sheets (CSS), and JavaScript for the front-end as well as Python 3.8 and its Flask application programming interface (API) with the integration of DeepSpeech 0.9.3 for the back-end. The app has been tested within different browsers on Ubuntu 20 and 22 as well as Windows 10 and 11 leveraging ffmpeg and sox for audio recording and conversion from opus-files into 16 kHz wav-files.

After pressing "A" on the keyboard for ATCo mode or "P" for pilot mode, the audio recording and live transcription starts. After releasing the pressed key, the audio file is saved with the current timetick in its filename. This timetick is reused for the transcription file name. The speech-to-text engine DeepSpeech continuously delivered the transcription of recognized words even if the utterance has not been completed yet, e.g., "false" in Figure 2 indicates that the endpoint of the utterance has not been reached as the push-to-talk button is not released yet.

The console version of the application is also able to use defined speech pauses as the endpoint for utterance recordings and their transcriptions. The audio filename, recognized words, and endpoint information are stored in a JavaScript object notation (JSON) file, which eases the readability by machines. as shown in the black bottom part of Figure 2.

As DeepSpeech offers to integrate own speech recognition models, the speech-to-text quality can be improved through utilizing sophisticated acoustic models and language models trained on ATC data. The application setup forms the lowest simulator fidelity category 'A' with the used voice recognition capability attributed to category 'C' of table 1 in [19].

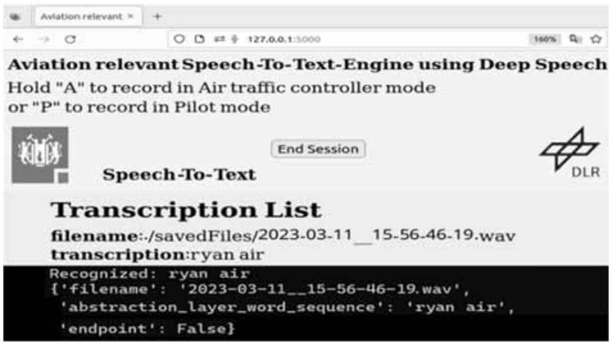


Figure 2: Screenshot-Collage of the Online-ATC-Speech-Recognition Interface with filename format, recognized word sequence, and endpoint information in graphical user interface (top) and JSON format (bottom).

2.3 ASRU Simulation Configuration

The ASRU module requires two important configuration inputs to be customized for a given ATC scenario. These inputs are set via configuration files in JSON format. The two files contain the ATC command types and the ATC concepts that shall or shall not be extracted for the given scenario.

Figure 3 shows three example entries of a JSON file for supported ATC command types and their qualifiers nested in the *Commands* array.

The *Type* key indicates the main part of the ATC command type while the *SndTypeName* indicates the sub part.

This allows to define command types such as *CLEARED LANDING, CLEARED ILS, TAXI TO,* or *VACATE VIA*. Some command types can have a *Qualifier* to specify the value such as *LEFT* or *EXPEDITE* for the command type *VACATE*.

The key *SupportedInThisAirspace* marks if the command type should be considered for the current simulation setup. This enables, e.g., to deactivate *PUSHBACK* commands for en route traffic scenarios (if *false*).

```
{ "Commands": [
 { "Type": "CLEARED",
   "SndTypeName": ["TAKEOFF","TOUCH_GO",
                   "LANDING","ILS",
                   "VISUAL","VIA","TO"] },
   "SupportedInThisAirspace": true,
 { "Type": "TAXI",
   "SndTypeName": ["TO","VIA"],
   "SupportedInThisAirspace": true },
 { "Type": "VACATE",
   "SndTypeName": ["VIA"],
   "Qualifier": ["LEFT","RIGHT","EXPEDITE"],
   "SupportedInThisAirspace": true }
]}
```

Figure 3: Configuration file excerpt example with supported ATC command types and qualifiers in JSON format.

Figure 4 shows two example entries of a JSON file for ATC concepts with further relevant values nested in the *AtcConcepts* array. The *Name* defines how an entity of an ATC concept shall be referred to. The *Locator*, e.g., contains a four-letter ICAO code for an airport such as *EYVI* for Vilnius where the concept is relevant.

```
{  "AtcConcepts": [
 { "Name": "JOZA",
   "Locator": "LHBP",
   "KeyWordSeq": ["joza","joza point",
                  "juliett oscar zulu alfa"],
   "CommandTypes": ["DIRECT_TO"],
   "ConceptType": "WAYPOINT",
   "AdditionalInfo":
      [{"LatLong": "47.592500 21.557222"}] },
 { "Name": "VILNIUS_TOWER",
   "Locator": "EYVI",
   "KeyWordSeq": ["tower vilnius","vilnius",
                  "vilnius airport",
                  "vilnius tower"],
   "CommandTypes": ["STATION"],
   "ConceptType": "FREQUENCY_POSITION",
   "AdditionalInfo":
   [{"FrequencyValues": "118.200"}] }  ]}
```

Figure 4: Configuration file excerpt example with ATC concepts, their word sequences, and additional information in JSON format.

The *KeyWordSeq* array lists all word sequences that should be mapped to the concrete ATC concept if they are found in an utterance transcription. The *CommandTypes* array lists all types that could make use of the ATC concept, i.e., a runway could e.g., be used for *CLEARED TAKEOFF* or *HOLD_SHORT*. The *ConceptType* represents the nature of the ATC concept. *AdditionalInfo* might contain numeral data about latitude/longitude of a waypoint or a frequency depending on the *ConceptType*.

With these two configuration files, it is possible to list the expected ATC commands and concepts. Hence, commands with Mach numbers can be excluded for tower scenarios and only those waypoints are added to the configuration file that exist in the current ATC environment. This enables to manually customize the command recognition, i.e., adapting speech understanding to the application without ASRU expert knowledge.

3 Multiple Remote Tower ASRU Simulations and Results

3.1 Real-Time Simulation with Controllers

For the HITL-RTS described in this paper, ATCos were responsible for three airports at the same time (named Vilnius, Kaunas, Palanga). The ATCos had three rows of monitors presenting the camera image of the respective airports and a head-down ATC system unit to monitor and influence the given traffic (see Figure 5). The communication with simulation pilots was done via radio telephony (over IP) on three different frequencies.

Figure 5: Multiple Remote Tower Setup in the Remote Tower Lab of DLR Braunschweig: One ATCo is controlling three airports, using head-down electronic flight strips and a tower radar display.

The ASRU module (1) automatically transcribed all ATCo utterances word-by-word and (2) automatically annotated the word sequences with the semantic meanings using a command extraction algorithm and the defined ontology of European ATC stakeholders.

The relevant recognized ATC commands were (3) displayed in an abstracted form in an ATCo display to be confirmed/maintained. An example transcription following defined transcription rules was: "*airest cargo five five zero vilnius tower you are cleared to destination via erlos one delta departure squawk is two one seven four startup approved QNH one zero two one runway one three*".

The relevant ATC commands with values were extracted from the transcription. The annotation of the above example transcription in a human-readable format, ignoring the JSON tagging, is:

AEG550 STATION VILNIUS_TOWER
AEG550 CLEARED TO DESTINATION
AEG550 CLEARED VIA ERLOS_1D
AEG550 STARTUP
AEG550 INFORMATION QNH 1021
AEG550 INFORMATION ACTIVE_RWY RW13.

The relevant recognized callsign and ATC command values of each utterance were automatically shown to the ATCo on an electronic flight strip display. This means, the aircraft with the recognized callsign was highlighted and the content of the ATC commands was displayed as abbreviated information either in text form or as symbols (e.g., an aircraft engine icon for STARTUP). The ATCo only needed to check the highlighted commands and correct if needed in seldom cases. Hence, these automatically entered and displayed information from verbal ATC commands reduced the manual ATCo workload for electronic flight strip maintenance.

Our conducted HITL-RTS evaluates the benefit of an ASRU module to support tower ATCos with electronic flight strip maintenance in a multiple remote tower environment. The setups for the HITL-RTS and pre-trials form the highest simulator fidelity category 'E' of table 1 in [19]. Hence, they were very realistic, but costly in the RTS conduction.

The verbal ATCo utterances of 116 roughly 45-minutes long RTS runs have been analysed in order to compare the automatic ATC concept extraction results with the actually intended ATC concepts. Therefore, all RTS runs have been automatically transcribed word-by-word and annotated concept-by-concept. Afterwards all of them were manually checked and corrected if necessary.

The HITL-RTS campaigns have been conducted at six different points in time with slightly varying setups between 2017 and 2022 with tower ATCos from four European ANSPs as follows: 17 from the Lithuanian ANSP, 13 from the Hungarian ANSP, 7 from the Austrian ANSP, and 3 from the Finnish ANSP. The complete data set is called "116/40" as it contains 116 simulation runs of 40 ATCos. It comprises 177,847 transcribed words with 32,436 commands in 10,712 audio files.

The simulation setup for the Lithuanian (LIT, 52 simulation runs), Hungarian (HUN, 41 simulation runs), and Austrian (AUT, 16 simulation runs) ATCos differed only very slightly in airport names. The aircraft callsigns, airport layouts, configuration files, etc. remained the same. However, the simulation setup for the Finnish (FIN, 7 simulation runs) ATCos included different callsigns, airport layouts, and configurations despite being a multiple remote tower (MRT) simulation with three remote airports and comparable traffic amount and traffic mix, too.

A subset of the complete data set, i.e., the final HITL-RTS with Lithuanian and Austrian ATCos in winter 2022 are analysed specifically. This data set contains ten simulation runs from Austrian ATCos and eight simulation runs from Lithuanian ATCos, in the latter eliminating two simulation runs of one ATCo due to technical issues, i.e., the sub-data set is called "18/9" due to 18 simulation runs of 9 ATCos. It comprises 35,022 transcribed words with 6963 commands in 2437 audio files.

3.2 Iterative ASRU Simulation Results

First, we present the results achieved with the ASRU model during and after the final simulation runs on the 18/9 data set, respectively. Second, we detail the ASRU results given the same ASRU model for the 116/40 data set. Third, we show the improved results with our current ASRU model on the 116/40 data set.

The following tables show the recognition rates (Recog) and error rates (Err) on callsign level (Csgn) and command level (Cmd) as well as the WER if applicable. The recognition and error rate results do not sum up to 100% due to not shown *rejection rate*, i.e., correctly annotated commands were not recognized at all, e.g., a startup, pushback, and taxi clearance were given, but only startup and taxi were recognized. Then we have one rejection. If the pushback would be replaced by, e.g., a climb command or the pushback value would be recognized wrongly, it as an error.

Table 1 shows the results for ASRU performance of three different speech recognition modes for the 18/9 data set. The mode *Live* shows the ASRU results on the continuous audio stream during the simulation runs as 'perceived' by the ATCos with a Kaldi based speech-to-text engine that has seen the earlier available audio files – before 2022 – as training data. This training data encompassed 3.6h of LIT and 0.9h of AUT next to other ATC data sources that did not match the final MRT simulation setup with Lithuanian and Austrian ATCos. The mode *AllTrain1* shows the WER for speech recognition on recorded wav-files with a Coqui speech-to-text engine and a model that has been trained on all available audio files after the final simulation runs (roughly 17h). The command and callsign rates are then computed on the speech-to-text output. The mode *Perfect1* considers manual transcriptions, assuming no errors. As expected, the WER is highest with least training data in *Live* mode. The more data, the better for ASRU performance. Therefore, the iterative approach is helpful to collect, and steadily faster transcribe and annotate more data for the next phase.

Speech Recognition Mode	Word Error Rate	Cmd Recog Rate	Cmd Err Rate	Csgn Recog Rate	Csgn Err Rate
Live	10.7	80.7	7.0	94.1	2.2
AllTrain1	3.2	92.0	4.2	98.4	0.7
Perfect1	0.0	95.6	2.9	99.7	0.2

Table 1: First results in [%] for speech recognition and understanding of HITL-RTS runs (18/9 data set).

Table 2 presents the speech understanding metrics based on the *AllTrain1* mode while Table 3 shows the results for the *Perfect1* mode on the 116/40 data set.

Data Set	Cmd Recog Rate	Cmd Err Rate	Csgn Recog Rate	Csgn Err Rate
All	93.1	4.1	98.4	0.8
MRT_HUN	93.8	4.3	98.3	1.0
MRT_LIT	94.2	3.4	98.7	0.7
MRT_AUT	90.6	5.1	97.5	0.9
MRT_FIN	85.1	5.2	98.6	0.8

Table 2: First results in [%] for ATC concept recognition on 116/40 data set given a WER of 3.2%.

Data Set	Cmd Recog Rate	Cmd Err Rate	Csgn Recog Rate	Csgn Err Rate
All	96.0	2.7	99.4	0.4
MRT_HUN	95.8	3.2	99.0	0.7
MRT_LIT	97.3	1.7	99.8	0.1
MRT_AUT	94.7	3.6	99.4	0.3
MRT_FIN	90.0	3.7	99.4	0.2

Table 3: First results in [%] for ATC concept recognition on 116/40 data set given a WER of 0%.

As expected, the higher WER in Table 2 leads to worse recognition rates and error rates on semantic level than in Table 3. However, with the WER of 3.2%, the callsign recognition only decreases by roughly 1% absolute and the command recognition decreases by 3% absolute only. This demonstrates that the speech understanding process can compensate a lot of word errors through the use of contextual data, due to redundant information in the utterances, and due to word errors, that affect irrelevant portions of a sentence in some cases.

Now, we present the most recent results given the available complete data set for training in our latest iteration of the process model. We created a first ASRU model *PartTrain* – this encompasses acoustic model, language model, and command extraction model – with training based on speech data, correct transcriptions, and correct annotations of Vienna approach. This model was applied on a multiple remote tower data test set resulting in a WER of 77%, a command recognition rate of 1%, and a callsign recognition rate of 24% (see Table 4). These results are useless even if the ASRU model performs acceptable when applying to Vienna approach data on which it has been trained with a WER of 6.2%, a command recognition rate of 85.6% (error rate 5.3%), and a callsign recognition rate of 96.9% (error rate 1.2%).

Speech Recognition Mode	Word Error Rate	Cmd Recog Rate	Cmd Err Rate	Csgn Recog Rate	Csgn Err Rate
PartTrain	77.0	1.3	8.5	24.3	34.7
AllTrain2	2.7	94.8	3.0	99.1	0.4
AllTune	1.8	95.7	3.1	99.3	0.6
Perfect2	0.0	97.1	2.2	99.5	0.3

Table 4: Current results in [%] for ATC concept recognition on 116/40 data set.

We created a second ASRU model *AllTrain2* – this encompasses acoustic model, language model, and command extraction model – with training based on speech data, correct transcriptions, and correct annotations of many different available ATC environments including two en route environments, three approach environments, and an apron environment as well as some multiple remote tower data. This model was applied on the complete multiple remote tower data – that was already part of the training data – resulting in a WER of 2.7%, a command recognition rate of 95% (error rate 3%), and a callsign recognition rate of 99% (error rate 0.4%) as shown in Table 4.

When using the improved command extraction model with enhancements for seldom used commands or new commands such as *STATION*, based on transcriptions with a WER of 0% in mode *Perfect2*, we achieve a command recognition rate of 97% (error rate 2.2%) and a callsign recognition rate of 99.5% (error rate 0.3%).

We created a third ASRU model *AllTune* – this encompasses acoustic model, language model, and command extraction model – with fine-tuning the first model *PartTrain* with the same data as for the second ASRU model *AllTrain2*. The *AllTune* model was applied on the complete multiple remote tower data – that was already part of the fine-tuning data – resulting in a WER of 1.8%, a command recognition rate of 96% (error rate 3.1%), and a callsign recognition rate of 99.3% (error rate 0.6%) as shown in Table 4.

If four out of five ATC commands as given with a command recognition rate of around 80% in Table 1 *Live* mode are automatically recognized and entered correctly into digital flight strips, this already saves manual effort of the ATCo to enter command content into the controller working position. This result was already achieved based on a WER of 11%.

With an even lower WER, a positive effect on the speech understanding metrics as outlined in Table 1 and 4 can be expected. For example, the recognition of command types, values, qualifiers, and conditions as calculated with a command recognition rate of 92.5% requires a manual correction by the ATCo in less than every 13th recognized command if ASRU output was visualized, e.g., in digital flight strips. This again, could translate into less ATCo workload, i.e., faster execution times for a secondary task, which can be interpreted as a higher availability of mental capacity of ATCos if they get ASRU support.

If the callsign error rate is below 1%, this means that less than every 100th callsign is wrongly recognized and, in case of callsign highlighting in an ATCo display, might rarely drag ATCo attention to an unintended spot. However, ASRU can enable to very often drag ATCo attention to the desired display spots.

Independent of the concrete ASRU result values having the same order of magnitude for other multiple remote tower or ATC setups in general, the ATCo tool support with given ASRU performance showed to be a valuable support for HITL-RTS simulations in the ATC domain.

4 Conclusion

We presented an iterative process, which enables to adapt existing speech recognition and understanding (ASRU) models to new environments, for which in the beginning no recorded training voice utterances exist. This, however, is a prerequisite to use ASRU support already during first human-in-the-loop simulations for new environments. First speech data for rough adaptation of existing ASRU models can be gained by the presented web-based online tool. Efforts for traveling and training as required for HITL-RTS itself were not necessary. The word error rate (WER) from untrained models of approximately 77% decreased to 11% in the case study using the described process model's first iterations. Data from initial training and verification runs can be used to iteratively fine-tune existing ASRU models for final simulation runs, which in turn improve the ASRU performance further.

The integration of ASRU lead to feasible ATCo support for ATC HITL-RTS. It supports ATCos maintaining aircraft information in electronic flight strips even with a WER of 11%, because the resulting command recognition rates of above 80% are already sufficient to free mental capacity for ATC tasks as shown through a secondary task. The performance difference of the secondary task with and without ASRU support has demonstrated that. Without integration of ASRU support already during first simulations, the results with respect to ATCo workload measurements might be useless, because ASRU support can reduce ATCos' workload by 20%.

Using all recordings from the 12,500 utterances of 116 simulation runs with 40 different ATCos for fine-tuning an ASRU model enabled a WER of 3% resulting in command recognition rates of 92-95% and callsign recognition rates of 99%.

Similar results can be expected for other new ATC environments modelled in simulators when using the presented iterative approach starting with recordings supported by a web-based tool.

Funding

One of the HITL-RTS was part of SESAR2020's PJ.05-W2 (Sol97), "Digital Tower Technologies (DTT)" that received funding from the SESAR Joint Undertaking under the European Union's Horizon 2020 research and innovation program under grant agreement No 874470.

References

[1] Dow C, Histon J. Enroute ATC Industry Perceptions of Simulation Fidelity. *18th International Symposium on Aviation Psychology*. May 2015, Dayton, OH, 524-529.

[2] Nguyen VN, Holone H. Possibilities, Challenges and the State of the Art of Automatic Speech Recognition in Air Traffic Control. 2015. DOI 10.5281/zenodo.1108428

[3] Helmke H, Kleinert M, Ohneiser O, Ahrenhold N, Klamert L, Motlicek P. Safety and Workload Benefits of Automatic Speech Understanding for Radar Label Updates. *Journal of Air Transportation* 32:4, 155-168, 2024. DOI 10.2514/1.D0419

[4] Ohneiser O, Helmke H, Shetty S, Kleinert M, Ehr H, Murauskas Š, Pagirys T, Balogh G, Tønnesen A, Kis-Pál G, Horváth V, Kling F, Rinaldi W, Mansi S, Piazzolla G, Usanovic H. Understanding Tower Controller Communication for Support in Air Traffic Control Displays. *12th SESAR Innovation Days*. Dec 2022, Budapest, Hungary.

[5] Kleinert M, Ohneiser O, Helmke H, Shetty S, Ehr H, Maier M, Schacht S, Wiese H. Safety Aspects of Supporting Apron Controllers with Automatic Speech Recognition and Understanding Integrated into an Advanced Surface Movement Guidance and Control System. *Aerospace*. 2023; 10, 596.

[6] Helmke H, Ondřej K, Shetty S, Arilíusson H, Simiganoschi TS, Kleinert M, Ohneiser O, Ehr H, Zuluaga JP, Smrz P. Readback Error Detection by Automatic Speech Recognition and Understanding. *12th SESAR Innovation Days*. Dec 2022, Budapest, Hungary.

[7] ICAO. Doc 4444 – Procedures for Air Navigation Services – Air Traffic Management. Montréal, QC, Canada: International Civil Aviation Organization; 16 ed., 2016.

[8] Badrinath S, Balakrishnan H. Automatic Speech Recognition for Air Traffic Control Communications. *Transportation Research Record*. 2022; 2676(1): 798-810.

[9] Lin Y. Spoken Instruction Understanding in Air Traffic Control: Challenge, Technique, and Application. *Aerospace*. 2021; 8, 65. DOI 10.3390/aerospace8030065

[10] Helmke H, Ohneiser O. Automatic Speech Recognition and Understanding in Air Traffic Management. *Aerospace*. 2024. DOI 10.3390/books978-3-7258-0315-6

[11] Ohneiser O, Helmke H, Shetty S, Kleinert M, Ehr H, Murauskas Š, Pagirys T. Prediction and extraction of tower controller commands for speech recognition applications. *J. Air Transp. Manag*. 2021; 95, 102089.

[12] Ohneiser O, Helmke H, Shetty S, Kleinert M, Ehr H, Schier-Morgenthal S, Sarfjoo S, Motlicek P, Murauskas Š, Pagirys T, Usanovic H, Meštrović M, Černá A. Assistant Based Speech Recognition Support for Air Traffic Controllers in a Multiple Remote Tower Environment. *Aerospace*. 2023; 10(6):560.

[13] Ohneiser O, Sarfjoo S, Helmke H, Shetty S, Motlicek P, Kleinert M, Ehr H, Murauskas Š. Robust Command Recognition for Lithuanian Air Traffic Control Tower Utterances. *InterSpeech*; Aug/Sep 2021; Brno, Czechia.

[14] Helmke H, Slotty M, Poiger M, Herrer DF, Ohneiser O, Vink N, Cerna A, Hartikainen P, Josefsson B, Langr D, García Lasheras R, Marin G, Mevatne OG, Moos S, Nilsson MN, Boyero Pérez M. Ontology for transcription of ATC speech commands of SESAR 2020 solution PJ.16-04. *Proceedings of the IEEE/AIAA 37th Digital Avionics Systems Conference (DASC).* Sep 2018; London, UK. DOI 10.1109/DASC.2018.8569238

[15] Helmke H, Shetty S, Kleinert M, Ohneiser O, Ehr H, Prasad A, Motlicek P, Cerna A, Windisch C. Measuring Speech Recognition and Understanding Performance in Air Traffic Control Domain Beyond Word Error Rates. *11th SESAR Innovation Days.* Dec 2021; Virtual.

[16] Ohneiser O, Helmke H, Ehr H, Gürlük H, Hoessl M, Mühlhausen T, Oualil Y, Schulder M, Schmidt A, Khan A, Klakow D. Air Traffic Controller Support by Speech Recognition. In: Ahram T, Karwowski W, Marek T, editors. *Advances in Human Aspects of Transportation: Part II.* Krakow, Poland: Applied Human Factors and Ergonomics (AHFE). 2014; 492-503.

[17] Taylor G, Miller J, Maddox J. Automating Simulation-Based Air Traffic Control. *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*. Nov/Dec 2005; 2913; Orlando, FL, USA.

[18] Zuluaga-Gómez JP, Prasad A, Nigmatulina I, Motlíček P, Kleinert M. A Virtual Simulation-Pilot Agent for Training of Air Traffic Controllers. *Aerospace*. 2023; 10(5). DOI 10.3390/aerospace10050490

[19] Dow C, Histon J. An Air Traffic Control Simulation Fidelity Definition and Categorization System. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. 2014; 58(1), 92-96.

[20] Meyer D, Wypych T, Petrovic V, Strawson J, Kamat S, Kuester F. An air traffic control simulator for test and development of airspace management schemes. *IEEE Aerospace Conference;* Mar 2018; Big Sky, MT, 2018.

[21] Kulkarni VB. Intelligent air traffic controller simulation using artificial neural networks. *International Conference on Industrial Instrumentation and Control (ICIC)*. May 2015; Pune, India, 1027-1031.

[22] Alam S, Abbass HA, Barlow M. ATOMS: Air Traffic Operations and Management Simulator. *IEEE Transactions on Intelligent Transportation Systems*. 2008; 9, 2, 209-225. DOI 10.1109/TITS.2008.922877

[23] Williams KW, Christopher B, Drechsler G, Pruchnicki S, Rogers JA, Silverman E, Gildea KM, Burian BK, Cotton S. Aviation Human-in-the-Loop Simulation Studies: Experimental Planning, Design, and Data Management. 2014; Washington, DC, FAA, TR DOT/FAA/AM-14/1.

[24] Chhaya B, Jafer S, Coyne WB, Thigpen NC, Durak U. Enhancing Scenario-Centric Air Traffic Control Training. *AIAA Modeling and Simulation Technologies Conference*. Jan 2018, Kissimmee, FL, USA.

[25] Fothergill S, Loft S, Neal A. ATC-lab^{Advanced}: An air traffic control simulator with realism and control. *Behavior Research Methods*. 2009; 41, 1, 118-127.

[26] Hoekstra J, Ellerbroek J. BlueSky ATC Simulator Project: An Open Data and Open Source Approach. *7th International Conference on Research in Air Transportation (ICRAT)*. Jun 2016, Philadelphia, PA, USA.

[27] García Lasheras R, Albarrán J, Fabio A, Celorrio F, Pinto de Oliveira C, Bárcena C. Automatic Flight Callsign Identification on a Controller Working Position: Real-Time Simulation and Analysis of Operational Recordings. *Aerospace.* 2023; 10, 433.

[28] Fridyatama DAS, Suparji S, Sumbawati MS. Developing Air Traffic Control Simulator for Laboratory. *TEM Journal*. 2023; 12, 3, 1462-1474, UIKTEN.

[29] Fürstenau N, Jakobi J, Papenfuss A. Introduction: Basics, History, and Overview. In: Fürstenau N, editor. *Virtual and Remote Control Tower Research Topics in Aerospace*. Cham, Switzerland: Springer; 2022; 3-22.

[30] Radford A, Kim JW, Xu T, Brockman G, McLeavey C, Sutskever I. Robust Speech Recognition via Large-Scale Weak Supervision. 2022. [Online] http://arxiv.org/pdf/2212.04356v1.

[31] Baevski A, Schneider S, Auli M. vq-wav2vec: Self-Supervised Learning of Discrete Speech Representations. *Proceedings of the 8th International Conference on Learning Representations, ICLR 2020*, Addis Ababa, Ethiopia, Apr 2020.

[32] Zuluaga-Gomez JP, Prasad A, Nigmatulina I, Sarfjoo S, Motlicek P, Kleinert M, Helmke H, Ohneiser O, Zhan Q. How Does Pre-trained Wav2Vec2.0 Perform on Domain Shifted ASR? An Extensive Benchmark on Air Traffic Control Communications. *Proceedings of the IEEE Spoken Language Technology Workshop (SLT).* Doha, Qatar, Jan 2023.

[33] May M, Kleinert M, Helmke H. Automatic Transcription of Air Traffic Controller to Pilot Communication – Training Speech Recognition Models with the Open Source Toolkit CoquiSTT. *DLRK Congress*, Hamburg, Germany, Sep 2024.

[34] Kleinert M, Venkatarathinam N, Helmke H, Ohneiser O, Strake M, Fingscheidt T. Easy Adaptation of Speech Recognition to Different Air Traffic Control Environments using the DeepSpeech Engine. *11th SESAR Innovation Days*. Dec 2021; Virtual.

[35] Zuluaga-Gomez JP, et al. ATCO2 corpus: A Large-Scale Dataset for Research on Automatic Speech Recognition and Natural Language Understanding of Air Traffic Control Communications. DOI 10.48550/arXiv.2211.04054

[36] Chen S, Kopald H, Tarakan R, Anand G, Meyer K. Characterizing National Airspace System Operations Using Automated Voice Data Processing A Case Study Exploring Approach Procedure Utilization. *Research and Development Seminar*. Vienna, Austria, Jun 2019.

[37] EUROCONTROL. ATC Mobility and Capacity Shortfalls. Think Paper #19 – 19 December 2022. [Online] http://s.dlr.de/atcmobilityandcapacityshortfalls.

Simulating a Pneumatics Network using the DLR ThermoFluidStream Library

Peter Junglas[1*], Raphael Gebhart [2]

[1]PHWT-Institut, PHWT Vechta/Diepholz, Am Campus 2, 49356 Diepholz, Germany; *peter@peter-junglas.de

[2]Institute of System Architectures in Aeronautics, German Aerospace Center (DLR), Münchener Str.20, 82234 Weßling, Germany

SNE 35(2), 2025, 117-124, DOI: 10.11128/sne.35.tn.10736
Selected ASIM WS 2025 Postconf. Publication: 2024-05-05
ReReceived Revised: 2025-05-15; Accepted: 2025-05-31
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Modeling and simulation of pneumatics networks is still a challenging task, plagued by initialization problems even in sophisticated environments such as the Modelica Fluid Library. The recently proposed DLR ThermoFluid Stream Library uses a promising new approach to cope with such problems. Therefore, it should be a convenient basis for a more specialized pneumatics library.

The essential concepts and components of such a library are presented, with a special focus on the notorious tee branch components. Their dynamic behaviour is very complex, since it couples the effects of dynamic pressure changes and friction losses, and often leads to stability problems. Results of systematic tests as well as more realistic models are discussed. They show that even though some problems with stability remain in special examples, the new library generally allows for the simulation of pneumatics networks using realistic tee branch models, which are more accurate than previous implementations.

Introduction

Modeling and simulation of pneumatic systems is a non-trivial endeavour, since it combines the turbulent flow of a compressible medium in a usually large pipe network with the highly non-linear behaviour of components such as actuators and valves [1]. A starting point for the mathematical description could be a non-linear partial differential equation describing the fluid flow, coupled with a set of ordinary differential equations modeling the mechanical behaviour of the corresponding components. Of course, this direct approach is usually unfeasible not only due to high computational demands, but because it requires a lot of fine-grained parameters to describe the model and provides much more data than is necessary for typical applications.

Applying a divide-and-conquer strategy, different modeling approaches are used according to the complexity of the system or component under study: A simple tee branch can be analyzed using the full power of a CFD simulation [2], while for more complex situations a coarse grained finite volume approach is employed, using discretizations in one or two dimensions [3]. To cope with very complex systems, one even reduces the description of many components to a zero-dimensional model, using ordinary differential or even purely algebraic equations to describe their behaviour, disregarding any spatial resolution. This approach is adopted in the Modelica Fluid library (“MFL”) [4] for most of its components.

For the modeling of large pneumatic networks the MFL has been used in [5]. Unfortunately, most models studied there didn’t run in standard Modelica environments, unless the behaviour of some components – especially the tee branches – had been simplified drastically. This is due to the structure of the model: For a pipe network the MFL approach leads to a large system of nonlinear equations, which needs very precise starting values to make the initialization converge.

The recently presented DLR ThermoFluidStream Library (“TFS”) [6] has been invented to address these problems. For this purpose, it adds the inertial pressure of the fluid, promoting the mass flows to state variables. Additionally, it uses a clever approximation scheme that decouples the equations of the components, without destroying the correct behaviour in static or quasi-static models [7]. Furthermore, the flows generally have fixed directions, which simplifies the modeling. As a consequence, the initialization usually works, even starting with vanishing mass flow, which should make it a suitable approach for the modeling of pneumatic pipe networks. It is the basis of the specialized PneuBibTFS library presented here, which is freely available from [8].

To show that the TFS library is up to this task, we will closely follow the lines of [5]: After a short introduction to the library and its basic components, a special focus will be on the modeling of tee branches, where several alternatives will be presented and extensively tested. Finally several variants of complete networks containing time-varying consumers will be analyzed and compared to the simplified versions presented in [5].

1 Using the DLR ThermoFluidStream Library

The problems with the initialization of models in different application areas are well-known, and a solution based on interpolating between the complete model and a simplified version has been proposed [9] and applied to thermofluid models [10]. Unfortunately, it only works in very special cases, especially not for pneumatic network models [5].

The TFS library addresses the initialization problem by introducing two major changes to the usual description of thermofluid models. They will be described briefly in the following, more details and motivations can be found in [7].

Integrating the Euler equation along a stream line leads to the "Newton's law like" pressure balance

$$\Delta r = \Delta q + \Delta p + \Delta p_{ext},$$

where Δq is the dynamic pressure difference due to change of velocity, Δp the pressure difference at the end points of the stream line, Δp_{ext} the pressure due to additional forces such as gravity or friction and Δr the pressure difference due to the inertia of the fluid, given by

$$\Delta r = -L\frac{d\dot{m}}{dt}.$$

The *inertance* L is independent of the thermodynamical state of a fluid and very small for gases. Since one is usually interested only in quasi-static processes, the inertial pressure difference Δr is neglected in the MFL library.

In the TFS library, models include this term, where L is generally defined as a globally set small value – since one is not really interested in the transient behaviour –, but can be set for each component individually.

The second ingredient of the TFS library is the introduction of the *steady mass flow pressure* $\hat{p}$, which is defined by splitting the total pressure as

$$p = \hat{p} + r.$$

Its change $\Delta\hat{p}$ along a stream line generally depends on the total pressure and the mass flow. In the steady state r vanishes, therefore the approximation

$$\Delta\hat{p} = f(p,\dot{m}) \approx f(\hat{p},\dot{m})$$

is generally sufficient for quasi-static simulations.

It leads to a decoupling of the component equations along the stream direction. This reduces the large set of nonlinear equations for the complete system to small-sized equations inside the components, thereby making the initialization problem feasible.

An important element in the design of a Modelica library is the connector. Instead of the stream connector used in the MFL library [11] the TFS library defines different connectors for ingoing and outgoing flows. They both use the mass flow $\dot{m}$ as flow variable and the initial pressure r as normal (*potential*) variable.

Additionally they contain the thermodynamic state as input or output variable, respectively. It is usually given by the pressure, the specific enthalpy and a set of mass fractions. Here, the steady mass flow pressure $\hat{p}$ is used instead of the total pressure, thereby implementing the approximation scheme described above.

Based on these ideas, the freely available TFS library contains many of the components that are needed for pneumatics simulations. The specialized pneumatics library PneuBibTFS mainly just contains wrappers around the TFS counterparts, which reduce the number of parameters to the few needed here, and fix the medium to SimpleAir. This further reduces possible non-linearities in the medium model, leading to enhanced stability.

Basic elements of PneuBibTFS generated in this way are:

- `Pipe`: a straight pipe with pressure loss according to Cheng [12].
- `Bend`: a curved pipe with pressure loss from the MFL dissipation library.
- `Tank`: an isothermal pressure tank with explicit inflow and outflow ports.

- `PressureSource`, `PressureSink`: simple source and sink with given pressure.
- `MassFlowSource`, `MassFlowSink`: source and sink that define an input or output mass flow. This is non-trivial in TFS, since the mass flow is a state variable. The components work by combining a pressure source or sink with a control valve from TFS that uses a PT1 dynamic to obtain the given mass flow.
- `MassFlowSourceLin`: source that uses a linear valve component to obtain a given input mass flow.
- `CVActuatorLin`: actuator using a linear valve to obtain a given output mass flow.

The linear mass flow source/sink components are simpler than their controlled counterparts and can lead to more stable models. Furthermore, they are used here to make results comparable to those of [5]. The critical tee branch components have to be created from scratch, they will be studied extensively in the following.

2 Modeling Tee Branches

As has been shown in [5], the modeling of the tee branch components is crucial for the stability of pneumatic network models.

This is mainly a consequence of their complex behaviour, combining pressure drops due to internal friction with dynamic pressure changes caused by the changed cross sections of the fluid flow.

In the case of splitting flows the division of the mass flows depends on incoming and outgoing pressures, which leads to a nonlinear coupling across the complete model. Using MFL-based components, even simple models did only run – i. e. survive the initialization phase –, when the tee branches were simplified drastically by completely disregarding all dynamical pressure changes.

Using the TFS approach instead, the mass flows become state variables, which breaks most of such loops. Since the flow directions are generally fixed in TFS, one now needs two different components: a splitter `TeeBranchS` and a junction `TeeBranchJ`, which join or split along the straight direction (cf. Fig. 1). For simplicity, we will only consider tee branches with a 90° angle and identical cross sections A at all three ports. This is a common situation in many pneumatics networks.

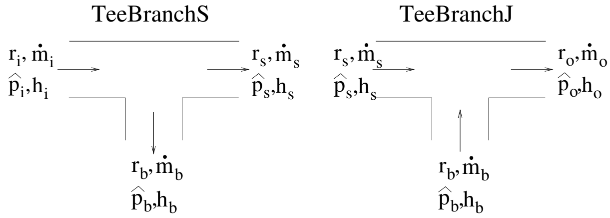


Figure 1: Tee Branch components.

The basic equations to describe the behaviour of a tee branch have been formulated in [13] and are widely used in applications. They rely on two functions ζ_{cs} and ζ_{cb} that describe the pressure losses across the straight and branch directions.

Since they contain a part of the dynamical pressure effects, they can be negative in certain cases, giving an actual pressure rise. Unfortunately, their concrete form varies largely in the literature [5]; we will use simple polynomials that fit published data.

The basic component `TeeBranchS` is simplified further by assuming constant temperature and density, using the density of the incoming flow everywhere. This avoids additional nonlinear loops inside the component and leads to the following equations:

$$
\begin{aligned}
0 &= \dot{m}_i + \dot{m}_s + \dot{m}_b \\
\rho &= \rho(\hat{p}_i, h_i) \\
\Delta p_s &= -\frac{1}{2\rho A^2}\zeta_{cs}\left(\frac{\dot{m}_b}{\dot{m}_i}\right)\dot{m}_i^2 \\
\Delta p_b &= -\frac{1}{2\rho A^2}\zeta_{cb}\left(\frac{\dot{m}_b}{\dot{m}_i}\right)\dot{m}_i^2 \\
\Delta p_{dyn,s} &= \frac{1}{2\rho A^2}(\dot{m}_i^2 - \dot{m}_s^2) \\
\Delta p_{dyn,b} &= \frac{1}{2\rho A^2}(\dot{m}_i^2 - \dot{m}_b^2) \\
\hat{p}_s &= \hat{p}_i + \Delta p_{dyn,s} + \Delta p_s \\
\hat{p}_b &= \hat{p}_i + \Delta p_{dyn,b} + \Delta p_b \\
h_s &= h_i \\
h_b &= h_i
\end{aligned}
$$

It is important to note that the equations to calculate the dynamic pressure differences Δp_{dyn} along the straight or branch direction are using the total input mass flow, while only a part of this mass flow reaches the corresponding output. This formulation is used, because the mass flow split is unknown beforehand.

The error introduced here is made up for by including the difference to the correct dynamical pressure in the ζ-functions – which makes clear, why they can have negative values.

These are the same equations that have been used in [5] for the split case, if one identifies $\hat{p}$ and p. In the TFS context, one needs additional equations describing the behaviour of the r variables. They can be derived from results in [7] or directly read off the component `SplitterN` provided in the TFS library:

$$L\ddot{m}_i = r_i - r_{mix}$$
$$L\ddot{m}_s = r_s - r_{mix}$$
$$L\ddot{m}_b = r_b - r_{mix}$$

where r_{mix} is an internal variable that is defined implicitely by the component equations.

A different approach to the modeling of a tee branch splitter uses the `DynamicSplitter` that is provided by the TFS library (cf. Fig. 2). It contains `DynamicPressureInflow/Outflow` components that compute dynamic pressure differences from the cross section area and the inlet/outlet velocity, which are given as parameter values. This leads exactly to the dynamic pressure differences from above.

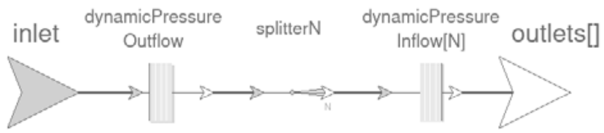


Figure 2: DynamicSplitter component.

The complete `TeeBranchS1` component adds a `SplitterPressureLoss` that computes the pressure loss caused by friction and the correction of the dynamical pressure, again using the ζ-functions (cf. Fig. 3). Basically, it reproduces the equations from above, with two small differences: The `DynamicPressureInflow/Outflow` include the temperature changes that are due to the – usually adiabatic, not isothermal – pressure change, and the frictional pressure computation uses the density at the outputs of the `DynamicSplitter`, not at the inlet. This corrects a part of the approximations that are made in the simpler `TeeBranchS`.

Furthermore, its approach is more modular and easier to understand.

On the other hand, its Modelica implementation consists of 147 equations altogether, compared to only 28 equations for the simpler component. Luckily, the Modelica preprocessing usually gets rid of this overhead.

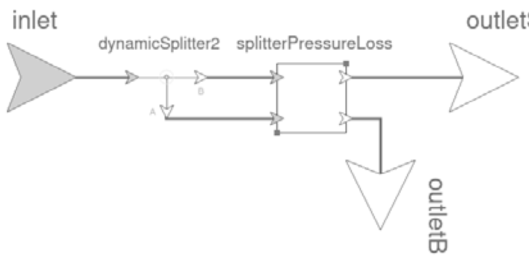


Figure 3: Alternative component TeeBranchS1.

To get even better results, one can use the component `TeeBranchS2`, which calculates the dynamic pressure differences by correctly using the densities of the input and output streams instead of using the input density everywhere.

The price is the addition of two nonlinear equations inside the component. A similar approach can be used with the `SplitterPressureLoss`, which would lead to two more nonlinear equations.

The construction of corresponding joining elements `TeeBranchJ`, `TeeBranchJ1` and `TeeBranchJ2` completely follows the lines above. The basic difference lies in the handling of the r variables when mixing input streams. The proper equations again can be found in [7] or in the component `JunctionN` from the TFS library.

3 Testing Tee Branches

The various tee branch components have been tested thoroughly using similar models as in [5], a typical example for the joining case is shown in Fig. 4. Here, the mass flows at the inflows and the pressure at the outflow are given explicitely.

The results for this example using the three different `TeeBranchJ` components and the `TeeBranch1` component from [5] are shown in Fig. 5. The plots for the basic MFL and TFS based components are almost identical, which is expected, since they use basically the same equations. The deviation at the beginning is due to the different initialization methods: MFL starts with a given value of $\dot{m}$, while TFS starts here with $\dot{m} = 0$ and winds it up using the inertance equation.

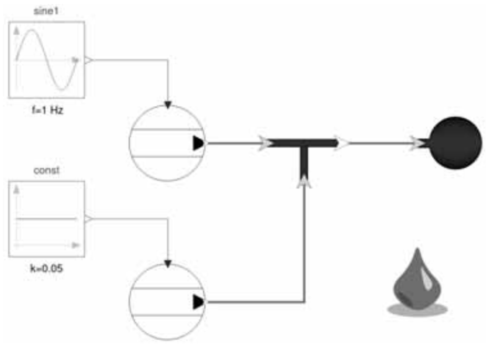


Figure 4: Model for testing a TeeBranchJ component.

The slight phase difference is not caused by the inertance, but by the PT1 dynamic of the mass flow controller used in TFS. Much larger are the pressure differences between the three TFS components, especially for the straight branch. At this point, this seems to indicate that a better modeling of the density changes could be useful.

More important than the exact results – which depend on the choice of the ζ-functions anyhow – is the question of stability: Do the models run immediately, only with special initial values or doesn't the initialization converge? To check this, test models similar to Fig. 4 have been analyzed, using different kinds of boundary conditions:

- a: pressure given at inflow, mass flow at outflow
- b: mass flow given at inflow, pressure at outflow
- c: pressure given at inflow and outflow

The results are unexpected: Only in the simple case, where two mass flows are given (case a for the splitter, case b for the joiner), all four components work. For the other cases, the MFL models work always, the basic TFS components in most cases, the more advanced TFS components never. The problem here is not the initialization, all models start and run for a (very) short time. Then the pressure values diverge rapidly. Obviously, the differential equations used here are highly unstable. In some cases, the problem can be fixed by using non-zero initial conditions for the mass flows, but often even very good starting points – coming from the working MFL model! – don't lead to a stable solution.

In additional tests a small pipe has been added either at the incoming or the outgoing straight branch. For the stable cases this leads to a problem with an MFL model: The splitter doesn't run with a pipe at output [5].

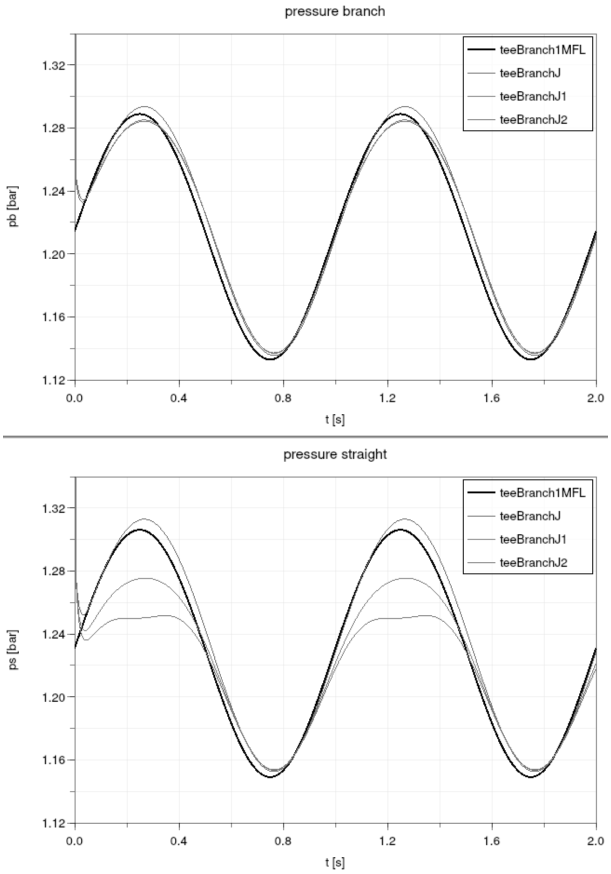


Figure 5: Comparison of the pressure drops in join mode.

The corresponding TFS models are not affected, they all work with the additional pipe on either side. In the unstable cases, the situation is more complicated, but generally, the situation gets worse in the MFL case, while in the TFS case several models that didn't run before, get stabilized by the additional pipe.

In conclusion, the tests show that the TFS approach does not solve all problems, due to the inherent instability of the basic equations. This apparently gets worse when the change in density is included. But at least it works in many cases, and the addition of pipes sometimes stabilizes a model.

4 Modeling Pneumatic Networks

To check the performance of the PneuBibTFS library in more realistic situations, the basic example model from [5] has been studied, which contains one `TeeBranchJ` and four `TeeBranchS` components, together with several pipes and curves, a pressure source, a few consumers and an auxiliary tank. Since the tank uses dedicated inflow and outflow ports, it is connected to the network via a loop consisting of a splitter and a joiner (cf. Fig. 6).

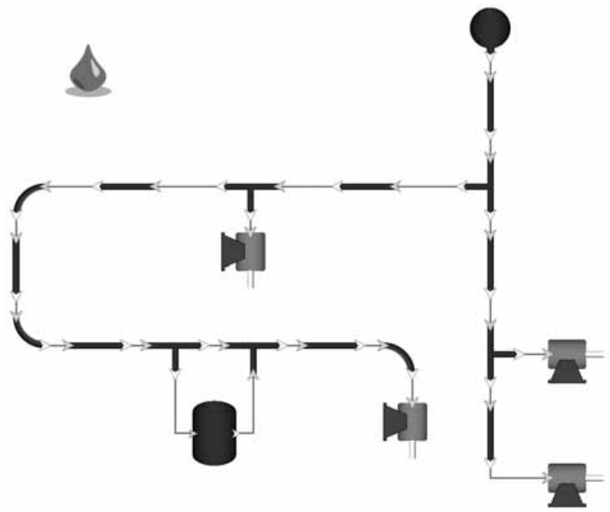


Figure 6: Model of the simple pneumatics network 1.

Starting with an empty tank (i. e. $p = p_0$), running the model works without problems and leads to results that are similar to those from [5] (cf. Fig. 7). If one replaces all tee branches by their more sophisticated versions, the models still run and reproduce the results of Fig. 7 within the plot accuracies. But in the MFS case, the model didn't run at all, unless one replaced the basic tee branch model by a very simplistic model based on substitutional pipe lengths. This shows that the somewhat unconvincing conclusions from the teebranch test results are much clearer in larger models: While the initialization problems in the MFL case get much more serious for larger models, in the TFS approach, the instabilities are largely mitigated.

A slightly extended example has been studied in [5] that contains an auxiliary tank between the two consumers on the right side. Building this model with PneuBibTFS, the simulation stops immediately with the error message

`Positive mass flow rate at Volume outlet.`

Apparently, in the initial phase of the simulation the medium flows into the tank through the outflow port, which is caught by an assertion. The TFS library includes a variable – hidden inside the `DropOfCommons` component – to reduce the assertion level from *error* to *warning*. Doing this, the model runs fine and produces the expected results. The backflow issue is a minor initialization problem and can be safely ignored here.

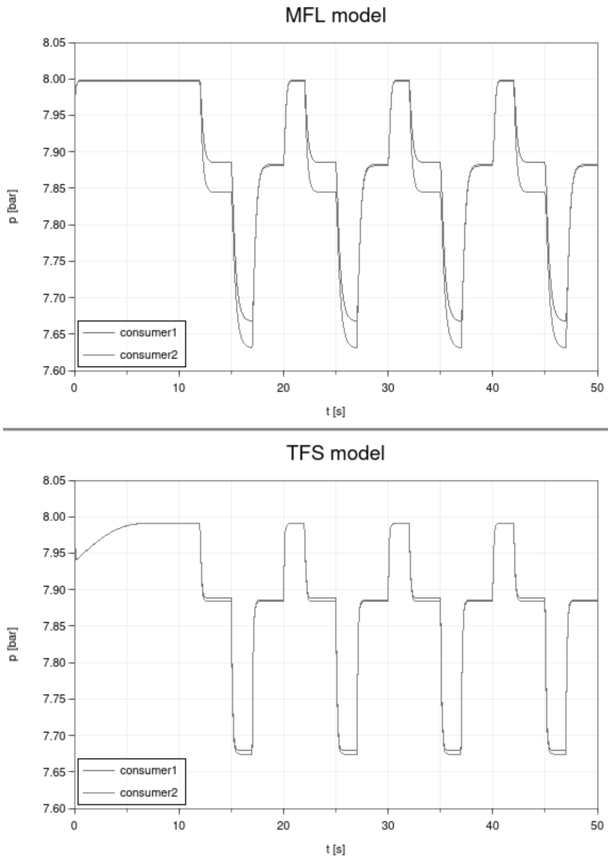


Figure 7: Simulation results of example network 1.

Increasing the simulation time one runs into another problem: The simulation stops at $t = 80$ s, one has hit the instability region. Taking a closer look at the model, one finds, that at this moment the consumer near the first tank is switched on for the first time. To increase the stability, a small pipe has been added between the splitter and the joiner that form the loop containing the tank (cf. Fig. 8). This works fine and the model now runs for long simulation times. Fig. 9 displays the pressure curves at the two consumers at the right side for the MFL and TFS variants. It shows clearly that the simplifications, which had been necessary to make the MFL model run, lead to significant deviations in the results.

Finally, one of the real-world models from [5], coming from an industrial partner, has been ported to PneuBibTFS. It contains almost 60 components, among them three pumps, one tank, 12 consumers and 17 tee branches.

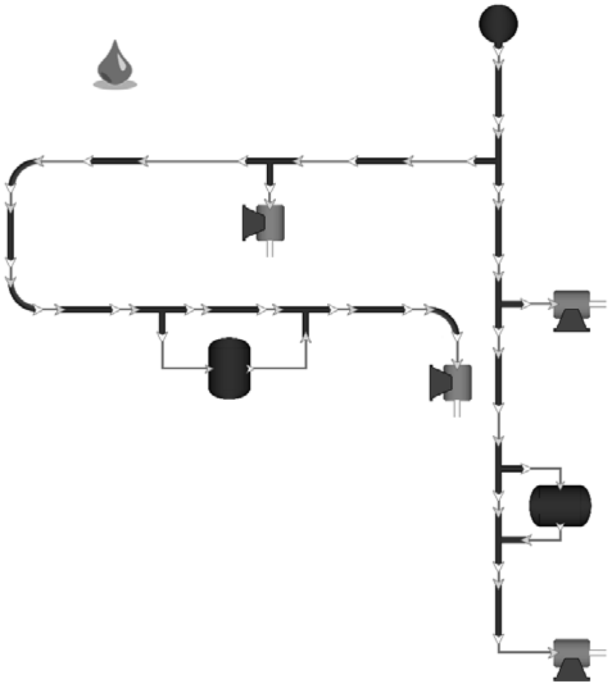


Figure 8: Model of the enlarged pneumatics network 2.

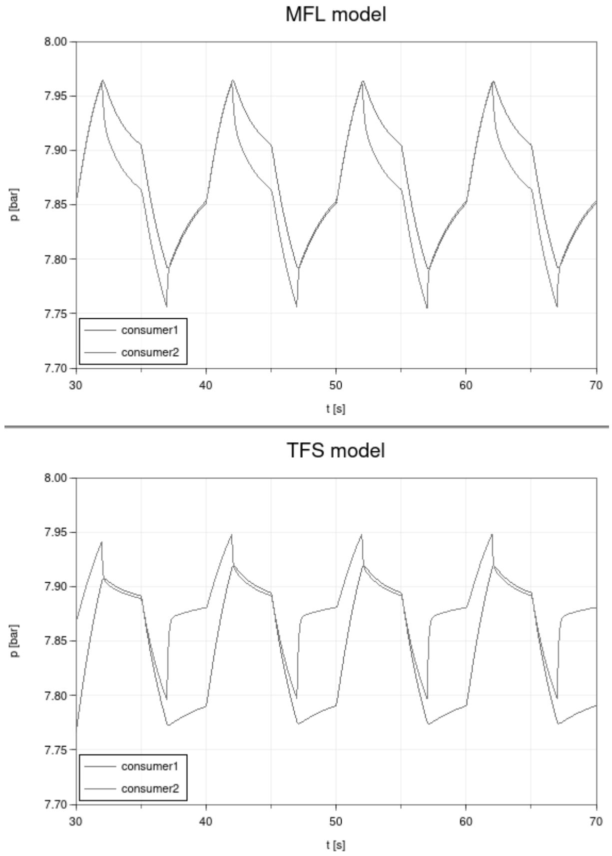


Figure 9: Simulation results of example network 2.

The MFL version only contains the simplistic tee branch component and has about 4500 equations. For the port to TFS the flow directions have to be specified everywhere. Furthermore, the tank again has to be included via a small loop, and its initial pressure has been set to the (identical) pump pressures. The final model has only 1750 equations, since the TFL library has a much simpler structure than the MFL library.

The simulation of the TFS-based model stopped after 1 s with the usual blowup of all pressures. Additionally, several flows had the wrong direction, which is due to the identical pressures of all pumps. To ensure the correct flow directions, the pressure of one pump has been increased marginally. With this change, the model runs immediately and qualitatively reproduces the results of the MFL version.

Comparison with the MFL-based results show significant differences, which are due to the very crude tee branch models used there. Apparently, the omission of the proper dynamic pressure changes introduced considerable errors.

Using the more detailed tee branch models that take into account local variations in density and temperature changed the results only marginally. Since these components reduce the general stability of the model, one should stick to the basic `TeeBranchS` and `TeeBranchJ` components.

5 Conclusions

Though the PneuBibTFS library still has problems with stability in special examples, it allows for the simulation of pneumatics networks using a realistic tee branch model. In many cases, the TFS-based methods work much better than an MFL-based approach, especially for larger models. If problems appear, they can often be cured by insertion of auxiliary pipes.

In [5] the use of OpenModelica [14] as a modeling and simulation tool introduced additional problems. This has changed completely, all PneuBibTFS models that run in Dymola [15], work in OpenModelica as well, and vice versa. This is due to two effects: On the one hand, the OpenModelica simulator has been enhanced considerably in the last years [16], on the other hand, the new models are much simpler conceptionally, since they don't lead to huge monolithic nonlinear equations . An interesting point for improvement is the model-

An interesting point for improvement is the modeling of the tanks: In reality, a tank is often connected to the pipe network using a simple port. It works as a buffer, the flow direction changes according to the pressure differences between the tank and the network. To model such a tank, one also needs a tee branch model that works with different flow directions. For such purposes, the TFS library has been enhanced to allow for bidirectional flows [17]. This leads to more complex components that are more tightly coupled. Whether such models deliver better results and – more importantly – are more stable, is an interesting question.

Clearly, the most important open point is the question of stability. Probably, the difficulties in solving the MFL-based nonlinear equations and the instability of the TFS-based differential equations are related. It would be interesting to study the instability of the basic tee branch equations in more detail and to find out, whether there exist more stable formulations, as well as how the stabilization in larger models actually works.

A basic conclusion from [5] with respect to the MFL library was: *The fundamental problem of initialization seems to be still far from being solved.*

In the light of the results presented here, it seems to be justified to claim that the TFS library has solved the initialization problem, at least for the class of models that have been studied here.

Publication Remark. This contribution is the improved version of the conference version published in Tagungsband Kurzbeiträge ASIM WS GMMS/STS 2025, ARGESIM Report AR 48,
ISBN ebook: 978-3-903347-66-3,
Volume DOI 10.11128/arep.48, p. 65-72.

References

[1] Beater P. *Pneumatic drives*. Berlin Heidelberg New York: Springer. 2007.

[2] Aigner D. Gleichung zur Berechnung der hydraulischen Verluste der Rohrvereinigung - kalibriert mit Ergebnissen numerischer und physikalischer Modelle. *3R-international*. 2008;47(1).

[3] Fischer F, Schmitz K. Distributed Parameter Pneumatics. In: *Proc. 15th Int. Modelica Conference*. Aachen, Germany. 2023; pp. 85–44.

[4] Franke R, Casella F, Sielemann M, Proelss K, Otter M. Standardization of thermo-fluid modeling in Modelica.Fluid. In: *Proc. 7th Int. Modelica Conference*. Como, Italy. 2009; pp. 122–131.

[5] Drente P, Junglas P. Simulating a simple pneumatics network using the Modelica Fluid library. *SNE Simulation Notes Europe*. 2015;25(2):85–92.

[6] Zimmer D, Weber N, Meißner M. The DLR ThermoFluid Stream Library. *Electronics*. 2022;11(22).

[7] Zimmer D. Robust object-oriented formulation of directed thermofluid stream networks. *Mathematical and Computer Modelling of Dynamical Systems*. 2020; 26(3):204–233.

[8] Junglas P. *Pneumatics Network library in Modelica*. URL `https://www.peter-junglas.de/fh/simulation/pneubib.html`

[9] Sielemann M, Casella F, Otter M, Clauß C, Eborn J, Mattsson SE, Olsson H. Robust Initialization of Differential-Algebraic Equations Using Homotopy. In: *Proc. 8th Int. Modelica Conference*. Dresden, Germany. 2011; pp. 75–85.

[10] Casella F, Michael Sielemann LS. Steady-state initialization of object-oriented thermo-fluid models by homotopy methods. In: *Proc. 8th Int. Modelica Conference*. Dresden, Germany. 2011; pp. 1–11.

[11] Franke R, Casella F, Otter M, Sielemann M, Elmqvist H, Mattson SE, Olsson H. Stream connectors – an extension of Modelica for device-oriented modeling of convective transport phenomena. In: *Proc. 7th Int. Modelica Conference*. 2009; pp. 108–121.

[12] Cheng NS. Formulas for friction factor in transitional regimes. *Journal of Hydraulic Engineering*. 2008; 134(9):1357–1362.

[13] Miller DS. *Internal Flow Systems*. Cranfield, UK: The British Hydromechanics Research Association, 2nd ed. 1990.

[14] Fritzson P, Pop A, Abdelhak K, Ashgar A, Bachmann B, Braun W, Bouskela D, Braun R, Buffoni L, Casella F, Castro R, Franke R, Fritzson D, Gebremedhin M, Heuermann A, Lie B, Mengist A, Mikelsons L, Moudgalya K, Ochel L, Palanisamy A, Ruge V, Schamai W, Sjölund M, Thiele B, Tinnerholm J, Östlund P. The OpenModelica Integrated Environment for Modeling, Simulation, and Model-Based Development. *Modeling, Identification and Control*. 2020;41(4):241–285.

[15] Brück D, Elmqvist H, Mattsson SE, Olsson H. Dymola for multi-engineering modeling and simulation. In: *Proc. 2nd Int. Modelica Conference*. Oberpfaffenhofen, Germany. 2002; pp. 55-1–55-8.

[16] Pop A, Östlund P, Casella F, Sjölund M, Franke R. A new OpenModelica compiler high performance frontend. In: *Proc. 13th Int. Modelica Conference*. Regensburg, Germany. 2019; pp. 689–698.

[17] Zimmer D, Weber N, Meißner M. Robust Simulation of Stream-Dominated Thermo-Fluid Systems: From Directed to Non-Directed Flows. *SNE Simulation Notes Europe*. 2021;31(4):177–184.

EUROSIM Data and Quick Info

Simulation in Production and Logistic - 21st ASIM Expert Conference
September 24-26, 2025, Dresden, Germany www.asim-gi.org/spl2025
ASIM SPL

SIMS
SIMS 2025
66th Int. Conference of Scandinavian Simulation Society – SIMS 2025
Stavanger, Norway, Sept. 23-24, 2025
www.scansims.org

Winter Simulation Conference 2025
December 7 - 10, 2025
Seattle, WA, USA www.wintersim.org

I3M 2025 **International Multidisciplinary Modeling & Simulation Multiconference**
Fes, Morocco, September 17-19, 2025 www.msc-les.org/i3m2025

EUROSIM Congress 2026
12th Congress of the European Simulation Societies
July 2026, Italy www.eurosim.info

EUROSIM – the **Federation of European Simulation Societies** was set up in 1989.

The purpose of **EUROSIM** is to provide a European forum for simulation societies and groups to promote modelling and simulation in industry, research, and development – by publication and conferences.

www.eurosim.info

EUROSIM members may be national simulation societies and regional or international societies and groups dealing with modelling and simulation.

Full Members are ASIM, CEA-SMSG, CSSS, DBSS, KA-SIM, LIOPHANT, LSS, PTSK, NSSM, SIMS, SLOSIM, UKSIM. Observer Members are ALBSIM and ROMSIM. Former Members (societies in re-organisation) are: CROS-SIM, FRANCOSIM, HSS, ISCS.

EUROSIM is governed by a B**oard** consisting of one representative of each member society, president, past president, and **SNE** representative.

President	Agostino Bruzzone (LIOPHANT) *agostino@itim.unige.it*
Past President	M. Mujica Mota (DBSS), *m.mujica.mota@hva.nl*
Secretary	Marina Massei (LIOPHANT), *massei@itim.unige.it*
Treasurer	Felix Breitenecker (ASIM) *felix.breitenecker@tuwien.ac.at*
Webmaster	Irmgard Husinsky (ASIM), *irmgard.husinsky@tuwien.ac.at*

SNE

SNE – Simulation Notes Europe is EUROSIM's membership journal with peer reviewed scientific contributions about all areas of modelling and simulation, including new trends as big data, cyber-physical systems, etc.

The EUROSIM societies distribute **e-SNE** in full version to their members as official membership journal. The basic version of **e-SNE** is available with open access (Creative Commons license CC BY). Publishers are **ASIM, ARGESIM** and **EUROSI.**

www.sne-journal.org
office@sne-journal.org

SNE-Editor: Felix Breitenecker (ASIM)
eic@sne-journal.org

EUROSIM Congress and Conferences

Each year a major **EUROSIM** event takes place, as the **EUROSIM CONGRESS** organised by a member society, **SIMS EUROSIM Conference**, and **MATHMOD Vienna Conference (ASIM)**.

On occasion of the **EUROSIM Congress 2023**, the *11th EUROSIM Congress* in Amsterdam, July, 2023, a new **EUROSIM** president has been elected: we welcome Agostino Bruzzone, well known simulationist, as new president. His society **LIOPHANT** will organize the next EUROSIM Congress in 2026 in Italy.

Furthermore, **EUROSIM Societies** organize local conferences, and **EUROSIM** co-operates with the organizers of **I3M Conference** and **WinterSim Conference Series.**

EUROSIM Member Societies

ASIM
German Simulation Society
Arbeitsgemeinschaft Simulation

ASIM is the association for simulation in the German speaking area, servicing mainly Germany, Switzerland and Austria.

President	Felix Breitenecker, *felix.breitenecker@tuwien.ac.at*
Vice President	Sigrid Wenzel, *s.wenzel@uni-kassel.de* Thorsten Pawletta, *thorsten.pawletta@hs-wismar.de* Andreas Körner, *andreas.koerner@tuwien.ac.at*

ASIM is organising / co-organising the following international conferences: **ASIM SPL** Int. Conference 'Simulation in Production and Logistics' (biannual), **ASIM SST** 'Symposium Simulation Technique' (biannual), **MATHMOD** Int. Vienna Conference on Mathematical Modelling (triennial). Furthermore, **ASIM** is co-sponsor of *WSC* - Winter Simulation Conference and of the *I3M* and conference series.

ASIM Working Committees

GMMS: Methods in Modelling and Simulation
U. Durak, umut.durak@dlr.de
SUG: Simulation in Environmental Systems
J. Wittmann, wittmann@informatik.uni-hamburg.de
STS: Simulation of Technical Systems
W. Commerell, commerell@hs-ulm.de
SPL: Simulation in Production and Logistics
S. Wenzel, s.wenzel@uni-kassel.de
EDU: Simulation and Education
A. Körner, andreas.koerner@tuwien.ac.at
Working Group Big Data: Data-driven Simulation in Life Sciences, N. Popper, niki.popper@dwh.at
Other Working Groups: Simulation in Business Administration, in Traffic Systems, for Standardisation, etc.

Contact Information

www.asim-gi.org

info@asim-gi.org, admin@asim-gi.org

ASIM – Office Germany, Univ. Bundeswehr Munich, Inst. für Technische Informatik, Tobias Uhlig, Werner-Heisenberg Weg 39, 85577 Neubiberg, Germany

ASIM – Office Austria, dwh Simulation Services, F. Breitenecker, N. Popper, Neustiftgasse 57-59, 1070, Wien, Austria

CEA-SMSG – Spanish Modelling and Simulation Group

CEA is the Spanish Society on Automation and Control. The association is divided into national thematic groups, one of which is centered on Modeling, Simulation and Optimization (**CEA-SMSG**).

President	José L. Pitarch, *jlpitarch@isa.upv.es*
Vice President	Juan Ignacio Latorre, *juanignacio.latorre@unavarra.es*

Contact Information

www.ceautomatica.es/grupos/

simulacion@cea-ifac.es

CEA-SMSG / Emilio Jiménez, Department of Electrical Engineering, University of La Rioja, San José de Calasanz 31, 26004 Logroño (La Rioja), Spain

CSSS – Czech and Slovak Simulation Society

CSSS is the Simulation Society with members from the two countries: Czech Republic and Slovakia. The CSSS history goes back to 1964.

President	Michal Štepanovský *michal.stepanovsky@fit.cvut.cz*
Vice President	Mikuláš Alexík, *alexik@frtk.fri.utc.sk*

Contact Information

cssim.cz

michal.stepanovsky@fit.cvut.cz

CSSS – Český a Slovenský spolek pro simulaci systémů, Novotného lávka 200/5, 11000 Praha 1, Česká republika

DBSS – Dutch Benelux Simulation Society

The *Dutch Benelux Simulation Society* (**DBSS**) was founded in July 1986 in order to create an organisation of simulation professionals within the Dutch language area.

President	M. Mujica Mota, *m.mujica.mota@hva.nl*
Vice President	A. Heemink, *a.w.heemink@its.tudelft.nl*
Secretary	P. M. Scala, *paolo.scala@fedex.com*

Contact Information

www.DutchBSS.org

a.w.heemink@its.tudelft.nl

DBSS / A. W. Heemink, Delft University of Technology, ITS – twi, Mekelweg 4, 2628 CD Delft, The Netherlands

KA-SIM Kosovo Simulation Society

The Kosova Association for Modeling and Simulation (**KA-SIM**) is closely connected to the University for Business and Technology (UBT) in Kosovo.

President	Edmond Hajrizi, *ehajrizi@ubt-uni.net*
Vice President	Muzafer Shala, *info@ka-sim.com*

Contact Information

www.ubt-uni.net

ehajrizi@ubt-uni.net

Dr. Edmond Hajrizi
Univ. for Business and Technology (UBT)
Lagjja Kalabria p.n., 10000 Prishtina, Kosovo

LIOPHANT Simulation

LIOPHANT Simulation is a non-profit association born in order to be a trait-d'union among simulation developers and users; **LIOPHANT** is devoted to promote and diffuse the simulation techniques and methodologies; the Association promotes exchange of students, sabbatical years, organization of International Conferences, courses and internships focused on M&S applications.

President	A.G. Bruzzone, *agostino@itim.unige.it*
Director	E. Bocca, *enrico.bocca@liophant.org*

Contact Information

www.liophant.org

info@liophant.org

LIOPHANT Simulation, c/o Agostino G. Bruzzone, DIME, University of Genoa, Savona Campus, via Molinero 1, 17100 Savona (SV), Italy

LSS – Latvian Simulation Society

The Latvian Simulation Society (LSS) has been founded in 1990 as the first professional simulation organisation in the field of Modelling and simulation in the post-Soviet area.

President	Artis Teilans, *Artis.Teilans@rta.lv*
Vice President	Oksana Kuznecova, *Oksana.Kuznecova@rtu.lv*

Contact Information

www.itl.rtu.lv/imb/

Artis.Teilans@rta.lv, Egils.Ginters@rtu.lv

LSS, Dept. of Modelling and Simulation, Riga Technical University, Kalku street 1, Riga, LV-1658, Latvia

NSSM – National Society for Simulation Modelling (Russia)

NSSM – The National Society for Simulation Modelling (Национальное Общество Имитационного Моделирования – НОИМ) was officially registered in Russia in 2011.

President	R. M. Yusupov, *yusupov@iias.spb.su*
Chairman	A. Plotnikov, *plotnikov@sstc.spb.ru*

Contact Information

www.simulation.su

yusupov@iias.spb.su

NSSM / R. M. Yusupov, St. Petersburg Institute of Informatics and Automation RAS, 199178, St. Petersburg, 14th line, h. 39

PTSK – Polish Society for Computer Simulation

PTSK is a scientific, non-profit association of members from universities, research institutes and industry in Poland with common interests in variety of methods of computer simulations and its applications.

President	Tadeusz Nowicki, *Tadeusz.Nowicki@wat.edu.pl*
Vice President	Leon Bobrowski, *leon@ibib.waw.pl*

Contact Information

www.ptsk.pl

leon@ibib.waw.pl

PSCS, 00-908 Warszawa 49, ul. Gen. Witolda Urbanowicza 2, pok. 222

SIMS – Scandinavian Simulation Society

SIMS is the Scandinavian Simulation Society with members from the five Nordic countries Denmark, Finland, Norway, Sweden and Iceland. The **SIMS** history goes back to 1959.

President	Tiina Komulainen, *tiina.komulainen@oslomet.no*
Vice President	Erik Dahlquist, *erik.dahlquist@mdh.se*

Contact Information

www.scansims.org

vadime@wolfram.com

Vadim Engelson, Wolfram MathCore AB,
Teknikringen 1E, 58330, Linköping, Sweden

SLOSIM – Slovenian Society for Simulation and Modelling

The Slovenian Society for Simulation and Modelling was established in 1994. It promotes modelling and simulation approaches to problem solving in industrial and in academic environments by establishing communication and cooperation among corresponding teams.

President	Goran Andonovski, *goran.andonovski@fe.uni-lj.si*
Vice President	Božidar Šarler, *bozidar.sarler@fs.uni-lj.si*

Contact Information

www.slosim.si

slosim@fe.uni-lj.si, vito.logar@fe.uni-lj.si

SLOSIM, Fakulteta za elektrotehniko, Tržaška 25,
SI-1000, Ljubljana, Slovenija

UKSIM - United Kingdom Simulation Society

The UK Modelling & Simulation Society (**UKSim**) is the national UK society for all aspects of modelling and simulation, including continuous, discrete event, software and hardware.

President	David Al-Dabass, *david.al-dabass@ntu.ac.uk*
Secretary	T. Bashford, *tim.bashford@uwtsd.ac.uk*

Contact Information

uksim.info

david.al-dabass@ntu.ac.uk

UKSIM / Prof. David Al-Dabass, Computing & Informatics, Nottingham Trent University, Clifton lane, Nottingham, NG11 8NS, United Kingdom

Observer Members

ROMSIM – Romanian Modelling and Simulation Society

ROMSIM has been founded in 1990 as a non-profit society, devoted to theoretical and applied aspects of modelling and simulation of systems.

Contact Information

florin_h2004@yahoo.com
ROMSIM / Florin Hartescu, National Institute for Research in Informatics, Averescu Av. 8 – 10, 011455 Bucharest, Romania

ALBSIM – Albanian Simulation Society

The Albanian Simulation Society has been initiated at the Department of Statistics and Applied Informatics, Faculty of Economy at the University of Tirana, by Prof. Dr. Kozeta Sevrani.

Contact Information

kozeta.sevrani@unitir.edu.al

Albanian Simulation Goup, attn. Kozeta Sevrani, University of Tirana, Faculty of Economy , rr. Elbasanit, Tirana 355, Albania

Former Societies / Societies in Re-organisation

- **CROSSIM** – *Croatian Society for Simulation Modelling*
  Contact: Tarzan Legović, *Tarzan.Legovic@irb.hr*
- **FrancoSim** – Société Francophone de Simulation
- **HSS** – Hungarian Simulation Society
  Contact: A. Gábor, *andrasi.gabor@uni-bge.hu*
- **ISCS** – Italian Society for Computer Simulation

The following societies have been formally terminated:

- **MIMOS** –Italian Modeling & Simulation Association; terminated end of 2020.

ASIM ASIM ASIM

ASIM Books – ASIM Book Series – ASIM Buchreihen

Monographs

Simulation-based Optimization: Industrial Practice in Production and Logistics
Lothar März, Markus Rabe, Oliver Rose (Eds.); to appear; ASIM Mitteilung 191

Energy-related Material Flow Simulation in Production and Logistics.
S. Wenzel, M. Rabe, S. Strassburger, C. von Viebahn (Eds.); Springer Cham 2023, print ISBN 978-3-031-34217-2, eISBN 978-3-031-34218-9, DOI 10.1007/978-3-031-34218-9, ASIM Mitteilung 182

Kostensimulation - Grundlagen, Forschungsansätze, Anwendungsbeispiele
T. Claus, F. Herrmann, E. Teich; Springer Gabler, Wiesbaden, 2019; Print ISBN 978-3-658-25167-3; Online ISBN 978-3-658-25168-0; DOI 10.1007/978-3-658-25168-0; ASIM Mitteilung 169

Proceedings*

Tagungsband ASIM Workshop 2025 - GMMS/STS - ASIM Fachgruppenworkshop 2025, DLR Oberpfaffenhofen
W. Commerell, U. Durak, D. Zimmer (Hrgs.), ARGESIM Report 48; ASIM Mitteilung AM 193
ISBN ebook 978-3-903347-66-3, DOI 10.11128/arep.48, ARGESIM Verlag Wien, 2025

Tagungsband Kurzbeiträge ASIM SST 2024 -27. ASIM Symposium Simulationstechnik, Univ. Bundeswehr München, München/Neubiberg, Sept. 2024, O. Rose, T. Uhlig (Hrgs.), ARGESIM Report 46; ASIM Mitteilung AM 189
ISBN ebook 978-3-903347-64-9, DOI 10.11128/arep.46, ARGESIM Verlag Wien, 2024

Tagungsband Langbeiträge ASIM SST 2024 -27. ASIM Symposium Simulationstechnik, Univ. Bundeswehr München München/Neubiberg, Sept. 2024,O. Rose, T. Uhlig (Hrgs.), ARGESIM Report 47; ASIM Mitteilung AM 190
ISBN ebook 978-3-903347-65-6, DOI 10.11128/arep.47, ARGESIM Verlag Wien, 2024

Simulation in Production and Logistics 2023 – 20. ASIM Fachtagung Simulation in Produktion und Logistik
TU Ilmenau, September 2023; S. Bergmann, N. Feldkamp, R. Souren, S. Straßburger (Hrsg.);
ASIM Mitteilung 187; ISBN ebook 978-3-86360-276-5, DOI: 10.22032/dbt.57476, Universitätsverlag Ilmenau, 2023

Proceedings Langbeiträge ASIM Workshop 2023 - STS/GMMS/EDU - ASIM Fachgruppenworkshop 2023
Univ. Magdeburg, März 2023; C. Krull; W. Commerell, U. Durak, A. Körner, T. Pawletta (Hrsg.)
ARGESIM Report 21; ASIM Mitteilung 185; ISBN ebook 978-3-903347-61-8, DOI 10.11128/arep.21, ARGESIM Verlag, Wien, 2023

Kurzbeiträge & Abstract-Beiträge ASIM Workshop 2023 STS/GMMS/EDU - ASIM Fachgruppenworkshop 2023
Univ. Magdeburg, März 2023; C. Krull; W. Commerell, U. Durak, A. Körner, T. Pawletta (Hrsg.)
ARGESIM Report 22; ASIM Mitteilung 186; ISBN ebook 978-3-903347-62-5, DOI 10.11128/arep.22, ARGESIM Verlag, Wien, 2023

Proceedings Langbeiträge ASIM SST 2022 -26. ASIM Symposium Simulationstechnik, TU Wien, Juli 2022
F. Breitenecker, C. Deatcu, U. Durak, A. Körner, T. Pawletta (Hrsg.), ARGESIM Report 20; ASIM Mitteilung AM 180
ISBN ebook 978-3-901608-97-1, DOI 10.11128/arep.20, ARGESIM Verlag Wien, 2022

Proceedings Kurzbeiträge ASIM SST 2022 -26. ASIM Symposium Simulationstechnik, TU Wien, Juli 2022
F. Breitenecker, C. Deatcu, U. Durak, A. Körner, T. Pawletta (Hrsg.), ARGESIM Report 19; ASIM Mitteilung AM 179
ISBN ebook 978-3-901608-96-4, DOI 10.11128/arep.19, ISBN print 978-3-901608-73-5, ARGESIM Verlag Wien, 2022

Book Series Fortschrittsberichte Simulation – Advances in Simulation *

An Architecture for Model Behavior Generation for Multiple Simulators. H. Folkerts, FBS 42
ISBN ebook 978-3-903347-42-7, DOI 10.11128/fbs.32, ARGESIM Publ. Vienna,2024

Das Verhalten von Transuranelementen in Erdböden - Theorie, Beprobung und radiochemische Analysen. K. Breitenecker, FBS 41; ISBN ebook 978-3-903347-41-0, DOI 10.11128/fbs.41, 2024; ISBN print 978-3-901608-99-5, 2010/2024; ARGESIM Publ. Vienna

Aufgabenorientierte Multi-Robotersteuerungen auf Basis des SBC-Frameworks und DEVS. B. Freymann, FBS 40
ISBN ebook2020_978-3-903347-40-3, DOI 10.11128/fbs.40, ARGESIM Publ. Vienna, 2022

Cooperative and Multirate Simulation: Analysis, Classification and New Hierarchical Approaches. I. Hafner, FBS 39
ISBN ebook978-3-903347-39-7, DOI 10.11128/fbs.39, ARGESIM Publ. Vienna,2022

Die Bedeutung der Risikoanalyse für den Rechtsschutz bei automatisierten Verwaltungsstrafverfahren. T. Preiß, FBS 38
ISBN ebook 978-3-903347-38-0, DOI 10.11128/fbs.38, ARGESIM Publ. Vienna,2020

Methods for Hybrid Modeling and Simulation-Based Optimization in Energy-Aware Production Planning. B. Heinzl, FBS 37
ISBN ebook 978-3-903347-37-3, DOI 10.11128/fbs.37, ARGESIM Publ. Vienna,2020;

Konforme Abbildungen zur Simulation von Modellen mit verteilten Parametern. Martin Holzinger, FBS 36
ISBN ebook 978-3-903347-36-6, DOI 10.11128/fbs.36, ARGESIM Publ. Vienna, 2020

Fractional Diffusion by Random Walks on Hierarchical and Fractal Topological Structures. G. Schneckenreither, FBS 35
ISBN ebook 978-3-903347-35-9, DOI 10.11128/fbs.35, ARGESIM Publ. Vienna, 2024

A Framework Including Artificial Neural Networks in Modelling Hybrid Dynamical Systems. Stefanie Winkler, FBS 34
ISBN ebook 978-3-903347-34-2, DOI 10.11128/fbs.34, ARGESIM Publ. Vienna, 2020

Modelling Synthesis of Lattice Gas Cellular Automata and Random Walk and Application to Gluing of Bulk Material. C. Rößler, FBS 33
ISBN ebook 978-3-903347-33-5, DOI 10.11128/fbs.33, ARGESIM Publ. Vienna, 2021

Combined Models of Pulse Wave and ECG Analysis for Risk Prediction in End-stage Renal Desease Patients. S. Hagmair, FBS 32
ISBN ebook 978-3-903347-32-8, DOI 10.11128/fbs.32, ARGESIM Publ. Vienna, 2024

Mathematical Models for Pulse Wave Analysis Considering Ventriculo-arterial Coupling in Systolic Heart Failure. S. Parragh, FBS 31
ISBN ebook 978-3-903347-31-1, DOI 10.11128/fbs.31, ARGESIM Publ. Vienna, 2024

Variantenmanagement in der Modellbildung und Simulation unter Verwendung des SES/MB Frameworks. A. Schmidt, FBS 30; ISBN ebook 978-3-903347-30-4, DOI 10.11128/fbs.30, ARGESIM Verlag, Wien 2019

Classification of Microscopic Models with Respect to Aggregated System Behaviour. Martin Bicher, FBS 29
ISBN ebook 978-3-903347-29-8, DOI 10.11128/fbs.29, ARGESIM Publ. Vienna, 2020

*** Download** via ASIM **www.asim-gi.org** Open Access - Basic Version Member Access - Enhanced Version