SIMUL 2019

The Eleventh International Conference on Advances in System Simulation

ISBN: 978-1-61208-756-6

November 24 - 28, 2019

Valencia, Spain

SIMUL 2019 Editors

Claus-Peter Rückemann, Leibniz Universität Hannover / Westfälische Wilhelms-Universität Münster, Germany

Jaime Lloret, Universitat Politecnica de Valencia, Spain

SIMUL 2019

Forward

The Eleventh International Conference on Advances in System Simulation (SIMUL 2019), held on November 24 - 28, 2019- Valencia, Spain, continued a series of events focusing on advances in simulation techniques and systems providing new simulation capabilities.

While different simulation events are already scheduled for years, SIMUL 2019 identified specific needs for ontology of models, mechanisms, and methodologies in order to make easy an appropriate tool selection. With the advent of Web Services and WEB 3.0 social simulation and human-in simulations bring new challenging situations along with more classical process simulations and distributed and parallel simulations. An update on the simulation tool considering these new simulation flavors was aimed at, too.

The conference provided a forum where researchers were able to present recent research results and new research problems and directions related to them. The conference sought contributions to stress-out large challenges in scale system simulation and advanced mechanisms and methodologies to deal with them. The accepted papers covered topics on social simulation, transport simulation, simulation tools and platforms, simulation methodologies and models, and distributed simulation.

We welcomed technical papers presenting research and practical results, position papers addressing the pros and cons of specific proposals, such as those being discussed in the standard forums or in industry consortiums, survey papers addressing the key problems and solutions on any of the above topics, short papers on work in progress, and panel proposals.

We take here the opportunity to warmly thank all the members of the SIMUL 2019 technical program committee as well as the numerous reviewers. The creation of such a broad and high quality conference program would not have been possible without their involvement. We also kindly thank all the authors that dedicated much of their time and efforts to contribute to the SIMUL 2019. We truly believe that thanks to all these efforts, the final conference program consists of top quality contributions.

This event could also not have been a reality without the support of many individuals, organizations and sponsors. We also gratefully thank the members of the SIMUL 2019 organizing committee for their help in handling the logistics and for their work that is making this professional meeting a success. We gratefully appreciate to the technical program committee co-chairs that contributed to identify the appropriate groups to submit contributions.

We hope the SIMUL 2019 was a successful international forum for the exchange of ideas and results between academia and industry and to promote further progress in simulation research.
We also hope Valencia provided a pleasant environment during the conference and everyone saved some time for exploring this beautiful city.

SIMUL 2019 Steering Committee

Hendrik Rothe, Helmut-Schmidt-Universität/ Universität der Bundeswehr Hamburg, Germany
Witold Pedrycz, University of Alberta, Canada
Ian Flood, University of Florida, USA

Marek Bauer, Politechnika Krakowska, Poland
Mu-Chun Su, National Central University, Taiwan
Etienne Kerre, Ghent University, Belgium
Kuan Yew Wong, Universiti Teknologi Malaysia (UTM), Malaysia
Yao Yiping, National University of Defence Technology - Changsha, Hunan, P. R. China

SIMUL 2019 Industry/Research Advisory Committee

Tsan-sheng Hsu, Institute of Information Science | Academia Sinica, Taiwan
Letizia Nicoletti, Cal-tek srl, Italy

SIMUL 2019 Publicity Chair

Ayman Aljarbouh, University of Grenoble Alpes (UGA) in Grenoble, France
Javier Rocher, Universitat Politecnica de Valencia, Spain

SIMUL 2019

Committee

SIMUL Steering Committee
Hendrik Rothe, Helmut-Schmidt-Universität/ Universität der Bundeswehr Hamburg, Germany
Witold Pedrycz, University of Alberta, Canada
Ian Flood, University of Florida, USA
Marek Bauer, Politechnika Krakowska, Poland
Mu-Chun Su, National Central University, Taiwan
Etienne Kerre, Ghent University, Belgium
Kuan Yew Wong, Universiti Teknologi Malaysia (UTM), Malaysia
Yao Yiping, National University of Defence Technology - Changsha, Hunan, P. R. China

SIMUL Industry/Research Advisory Committee
Tsan-sheng Hsu, Institute of Information Science | Academia Sinica, Taiwan
Letizia Nicoletti, Cal-tek srl, Italy

SIMUL Publicity Chair
Ayman Aljarbouh, University of Grenoble Alpes (UGA) in Grenoble, France
Javier Rocher, Universitat Politecnica de Valencia, Spain

SIMUL 2019 Technical Program Committee

Kareem Abdelgawad, Audi AG - Ingolstadt, Germany
El-Houssaine Aghezzaf, Ghent University, Belgium
Reza Akhavian, California State University East Bay, USA
Mikulas Alexik, University of Zilina, Slovak Republic
Ayman Aljarbouh, University of Grenoble Alpes (UGA) in Grenoble, France
Ciro Alberto Amaya, University of Los Andes, Bogotá, Colombia
Frédéric Amblard, IRIT | Université Toulouse 1 Capitole, France
Boris Andrievsky, Russian Academy of Sciences, Russia
Chrissanthi Angeli, Piraeus University of Applied Sciences, Greece
Amr Arisha, Dublin Institute of Technology (DIT), Ireland
Alfonso Ariza Quintana, University of Malaga, Spain
Marek Bauer, Politechnika Krakowska, Poland
Paul-Antoine Bisgambiglia, University of Corsica Pasquale Paoli, France
Paolo Bocciarelli, University of Rome Tor Vergata, Italy
Matias Bonaventura, FCEyN, UBA / ICC, CONICET, Argentina
Jalil Boudjadar, Aarhus University, Denmark
Christos Bouras, University of Patras | Computer Technology Institute & Press «Diophantus», Greece
Benjamin Camus, Inria | IRISA, Rennes, France
Paula M. Castro Castro, Universidade da Coruña, Spain
Vanessa I. Cedeño-Mieles, Escuela Superior Politécnica del Litoral (ESPOL), Guayaquil, Ecuador
Eugene Ch'ng, University of Nottingham Ningbo, China

Copyright Information

For your reference, this is the text governing the copyright release for material published by IARIA.

The copyright release is a transfer of publication rights, which allows IARIA and its partners to drive the dissemination of the published material. This allows IARIA to give articles increased visibility via distribution, inclusion in libraries, and arrangements for submission to indexes.

I, the undersigned, declare that the article is original, and that I represent the authors of this article in the copyright release matters. If this work has been done as work-for-hire, I have obtained all necessary clearances to execute a copyright release. I hereby irrevocably transfer exclusive copyright for this material to IARIA. I give IARIA permission or reproduce the work in any media format such as, but not limited to, print, digital, or electronic. I give IARIA permission to distribute the materials without restriction to any institutions or individuals. I give IARIA permission to submit the work for inclusion in article repositories as IARIA sees fit.

I, the undersigned, declare that to the best of my knowledge, the article is does not contain libelous or otherwise unlawful contents or invading the right of privacy or infringing on a proprietary right.

Following the copyright release, any circulated version of the article must bear the copyright notice and any header and footer information that IARIA applies to the published article.

IARIA grants royalty-free permission to the authors to disseminate the work, under the above provisions, for any academic, commercial, or industrial use. IARIA grants royalty-free permission to any individuals or institutions to make the article available electronically, online, or in print.

IARIA acknowledges that rights to any algorithm, process, procedure, apparatus, or articles of manufacture remain with the authors and their employers.

I, the undersigned, understand that IARIA will not be liable, in contract, tort (including, without limitation, negligence), pre-contract or other representations (other than fraudulent misrepresentations) or otherwise in connection with the publication of my work.

Exception to the above is made for work-for-hire performed while employed by the government. In that case, copyright to the material remains with the said government. The rightful owners (authors and government entity) grant unlimited and unrestricted permission to IARIA, IARIA's contractors, and IARIA's partners to further distribute the work.

Table of Contents

Construction Equipment Emission Modeling and Activity Analysis Using Deep Learning

Reza Akhavian
Department of Civil, Construction, and Environmental Engineering
San Diego State University
San Diego, United States
Email: rakhavian@sdsu.edu

***Abstract*—Automated activity recognition and modeling of heavy construction equipment can contribute to the correct and accurate measurement of a variety of project performance indicators. Productivity assessment and sustainability measurement through equipment activity cycle monitoring to eliminate ineffective and idle times thus reducing Greenhouse Gas (GHG) Emission, are some potential areas that can benefit from the integration of automated activity recognition and analysis techniques. In light of this, this idea paper describes design and development of a deep-learning framework that uses accelerometer data to detect activities of construction equipment and consequently estimates the emission produced corresponding to various activities.**

Keywords-construction equipment; machine learning; emission; modeling; sensors .

I. INTRODUCTION

Thanks to the cost-effective, ubiquitous, and computationally powerful means of data collection and analysis, data-informed process mining and decision making have become prevalent. The Architectural, Engineering, Construction, and Facility Management (AEC/FM) industry as well, begins to realize the benefits of such approaches.

Monitoring construction resources, such as heavy equipment enables not only improvements in productivity but also increased awareness of emissions produced as a result of fuel consumption. Studies conducted by the United States Environmental Protection Agency (EPA) demonstrate that heavy-duty construction equipment is one of the major contributors of emissions from diesel engines [1].

In order to reduce emissions, it is practical to reduce the time that construction equipment spends doing non-value-adding activities and/or idling. Research indicates that although using newer equipment, using well-maintained equipment, and using clean fuels can improve exhaust emissions, reducing engine idling time and enhancing equipment operating efficiencies results in improved outcomes [2][3]. Therefore, timely evaluation and monitoring of key construction activities may significantly contribute to foresee potential failures prior the project execution phase.

The ultimate goal of this line of research is to develop a model that enhances sustainability measures of construction projects by monitoring the operational efficiency and environmental performance of working equipment. Direct means of measurement, such as portable emission measurement equipment systems (PEMS) provide a reliable means of quantifying emissions. While PEMS provide direct and reliable measurements, use of PEMS in routine emissions assessments is questionable since they require time-consuming setup, calibrations, and data collection.

This idea paper explores the feasibility of developing an automated system for construction equipment analysis that helps monitor activities by leveraging sensors, such as inertial measurement units (IMUs) and deep learning technologies. The ultimate goal is to provide a detailed equipment operational analysis smart enough to detect idle times and the performance of no-value added activities. This approach is less time consuming, expensive, and error-prone compared to the manual methods.

The rest of this paper is structured as follows. In Section II, the methodology adopted is introduced and Section III discusses the conclusion and future directions of this work.

II. PROPOSED METHODOLOGY

Previous work by the author has shown that end-to-end deep learning models can learn to accurately classify the activities of construction equipment based on vibration patterns picked up by accelerometers attached to the equipment [4]. The work proposed here extends this prior work in two ways: (1) it introduces a new architecture that simplifies the previous approach while improving activity classification performance and (2) it explicitly studies relationships between the equipment activities and the emissions that they generate.

A. Data

Two of the datasets mentioned in this work for comparison purposes were collected and originally studied in a prior work led by the author [4]. These datasets study a BOMAG BW 145PDH-3 compactor performing six different activities and a CAT 328D excavator (Excavator 1) performing seven activities. In each experiment, the equipment was outfitted with two 3-axis accelerometers mounted at different locations, which produced six channels worth of vibration patterns. The data were manually labeled with their corresponding activity classifications according to video taken of the experiments, separated into disjoint training and validation sets, segmented into smaller sequences using sliding windows, and used to train two deep learning models.

 ISBN: 978-1-61208-756-6

This work uses the same approach to data collection as these previous studies, but this time a CAT 305D CR excavator (Excavator 2) performing seven different activities was studied and, in addition to the accelerometer readings, emissions data for various gases were collected using a PEMS. In total, 377,808 accelerometer readings were collected at a sampling rate of 100 Hz. Because the PEMS operated at a sampling rate of 1 Hz, its readings were upsampled to 100 Hz to match those of the accelerometers. The first 324,579 readings (85.9%) were used as training data while the remaining 53,229 readings (14.1%) were used for validation of the results. This split was chosen so as to ensure similar activity distributions in the training and validation sets.

The emissions signals collected and studied include carbon monoxide (CO), nitrogen oxides (NO_x), and carbon dioxide (CO_2). Because the carbon dioxide emissions were much larger, they are reported on a percentage scale, while the other signals are reported in parts-per-million (ppm). Figure 1 plots the emission signals vs. time below.

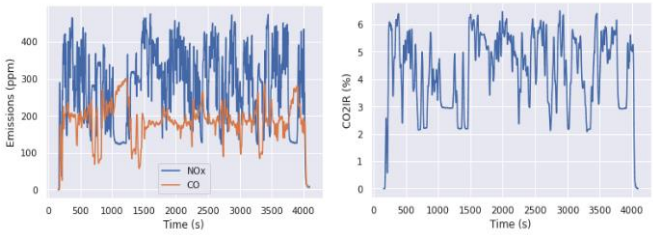


Figure 1. CO, NO_x, and CO_2 emissions vs. time.

B. *Predicting Equipment Activities from the Accelerometer Readings*

The author applied the same data processing techniques as in prior work. That is, the readings in each sensor channel were normalized to fall into the range [0, 1] and segmented into training examples 200 time steps x 6 sensor channels each using an overlapping sliding window process. Each frame was labeled according to the activity at the last time step so as to structure the problem as follows: given the 200 most recent accelerometer readings across six channels, predict the activity label at the 200th time step.

C. *Linking Equipment Activities to Emissions*

For each emissions signal considered, the readings were separated by activity and plotted as histograms in order to estimate their true distributions. The training and validation subsets of the data were considered separately.

D. *Deep Learning Architecture*

A *BaselineCNN* and a *DeepConvLSTM*, two models adapted for construction equipment activity recognition based on models of the same names originally developed for human activity recognition by [5] are investigated. Both *BaselineCNN* and *DeepConvLSTM* begin with four layers of convolutional filters meant to automatically extract features from the accelerometer time series. *BaselineCNN* then uses two fully-connected layers to interpret these extracted features and make a classification while *DeepConvLSTM* uses two Long Short-term Memory (LSTM) layers to interpret the extracted features and make its own classification. LSTMs are a particularly popular and performant kind of recurrent neural network (RNN), which are a broad class of networks that deal with sequence data.

Temporal convolutional networks (TCNs) are another kind of network designed to deal with sequence data. Traditional convolutional networks (CNNs) are suited to extracting locally correlated features but not suited to interpreting features that are distant from each other in space or time. This is because the receptive field of a convolutional network scales linearly with its number of layers. In order to achieve a larger receptive field that scales exponentially with the number of convolutional layers, the concept of dilated convolutions to CNNs is applied.

III. CONCLUSION AND FUTURE WORK

This Idea Paper proposes the use of deep learning algorithms for activity recognition of construction equipment as well as quantifying the emission attributed to activities. Preliminary analysis reveals promising results to predict the emission associated with activities. It was seen, across all of the measurements taken, that the new *TCN* model is at least competitive with the previous reigning champion, *DeepConvLSTM*. In fact, it beats *DeepConvLSTM* in terms of validation accuracy every time, despite training much faster and being simpler to explain. In the first excavator experiment, *DeepConvLSTM* managed a validation accuracy of 82.5%, but the *TCN* managed to achieve 83.4% validation accuracy regardless of whether the *Various* label was present. In the second excavator experiment the *TCN* achieved a validation accuracy of 78.8%. The ongoing and future directions of this work include extending the deep learning model created for activity recognition to also predict the emission levels.

REFERENCES

[1] EPA, OAR US. *Recommendations for Reducing Emissions from the Legacy Diesel Fleet*. 10 17, 2014 Available from: https://www.epa.gov/caaac/recommendations-reducing-emissions-legacy-diesel-fleet. [accessed October 15, 2019]

[2] C. Ahn,, S. H. Lee, and F. Peña-Mora, "Carbon Emissions Quantification and Verification Strategies for Large-scale Construction Projects," in Proc. Int. Conf. on Sustainable Design and Construction, pp. 1-18, 2013.

[3] R. Akhavian and A. H. Behzadan, "Simulation-Based Evaluation of Fuel Consumption in Heavy Construction Projects by Monitoring Equipment Idle Times," in Simulation Conference (WSC), pp. 3098-3108, 2013.

[4] C. Hernandez, T. Slaton, V. Balali, and R. Akhavian, "A Deep Learning Framework for Construction Equipment Activity Analysis," in Computing in Civil Engineering: pp. 479-486, 2019.

[5] F. Ordóñez and D. Roggen, "Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition," Sensors, 16(1), 115, 2016.

 ISBN: 978-1-61208-756-6

A Reactive "In silico" Simulation for Theoretical Learning Clinical Skills and Decision-Making

Alex Vicente-Villalba, Montserrat Antonin,
Escuelas Universitarias Gimbernat: Nursing School
Universitat Autònoma de Barcelona (UAB)
Barcelona, Spain
e-mail: alejandro.vicente@eug.es,
montserrat.antonin@eug.es

Dolores Rexachs, Emilio Luque
Computer Architecture and Operating Systems Department
Universitat Autònoma de Barcelona (UAB)
Barcelona, Spain
e-mail: Dolores.rexachs@uab.cat,
emilio.luque@uab.cat

***Abstract*- One of the problems found nowadays in the field of healthcare, and in particular in the emergency services, is that clinical staff members have to make many decisions and they have to implement them promptly. In addition, in the academic field, we must work to train in the competence of decision making with the aim of improving both critical thinking skills and clinical skills. The aim of our research is to develop a conceptual model of the evolution of the Chronic Obstructive Pulmonary Disease (COPD) in a patient, depending on the performances (INPUTS) of the clinician or student (medicine/nursing), in order to transfer that knowledge to a VIRTUAL computerized model. The simulator will enhance a student's training in decision-making and the assessment of acquired skills. Clinical variables, conditions, the evolution of the disease and the input of the clinicians who attend the patient in emergency services define the model. Considering the system configuration, the training offered will be more or less controlled depending on the expertise of the student. Moreover, it will show them not only those scenarios which are more frequent probabilistically, but also those scenarios which are less usual but likely to be encountered in real life.**

Keywords-Simulation; Nursing and Medical Education; Learning; computer models.

I. INTRODUCTION

Chronic pathologies, especially Chronic Obstructive Pulmonary Illness (COPD), are very common in our society and they are responsible for a high number of visits to our emergency services. Correct decision making can result in chronic-pathology sufferers making fewer visits to emergency departments. Nowadays, being capable in the professional field consists of solving problems and situations at work in an autonomous manner. All the knowledge acquired in the formative stages, whether it is higher education or professional education, is not enough since it is essential to have the skills, knowledge, and attitude needed to perform the specific tasks of a job.

Today, unfortunately, medicine and nursing students can learn a great deal about physiopathology, pharmacology, specific care, but they may ignore, or not know, how to use all that knowledge in a stressful situation, such as working in the emergency department.

Drawing from our own career, we have concluded that this deficiency is due to the decontextualization created by formal learning. In the same way, we see that not only students, but professionals as well must, on many occasions, make decisions and implement them as quickly as possible. This many sometimes lead to error due to the inability to assess multiple scenarios quickly when situations are unexpected. If we refer to the mortality data included in the report published in 1999 by the Institute of Medicine of the United States and titled "To Err is Human: building a safer health system", nearly 100,000 deaths were estimated to have been caused by medical errors [1]. Hence, back then, the need to improve the training of the professionals in order to avoid these errors was had already been suggested. The increase in medical errors was blamed on several factors, some of them being the lack of investment in technology and the growing complexity therapeutic procedures. As a result of such report, health educators started to add simulation components to their educational activities.

As stated by Vázquez and Guillamet [2], it has been shown that the use of these clinical simulations reduces the time needed to learn the skills. Figure 1 shows Miller's pyramid, where in the DEMONSTRATE and DO levels the learning processes in simulation are found [3].

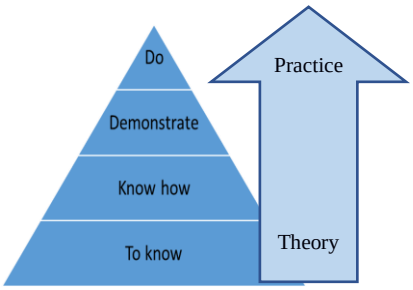


Figure 1. Miller's pyramid

Furthermore, it is claimed that the learning curves based on simulation are better than the curves based on classical learning [4].

As opposed to the potential shown by the implementation of clinical simulation in the degree or professional field, this implementation is not without its challenges. Clinical simulation is not a technology but a technique or learning methodology focused on the

 ISBN: 978-1-61208-756-6

participant. Therefore, the main limitation to its widespread application is the high price resulting from its formation in teaching methodology, infrastructure and the excessive amount of time spent by them and the participants in the clinical activity [5].

If we concentrate deeply on the world of simulation, the use of computers encouraged the use of mathematics in the development of physiology and pharmacology, communication around the world and the design of virtual worlds similar to reality [6]. Thanks to this, the development of software to train in the medical field (doctors and nurses) has been made possible. In this software, the user was depicted as a medical or nursing professional and was able to work with virtual tools and perform therapeutic procedures [7].

Authors such as Aldrich [8], highlight that computational simulation is a genre that can help students since it allows self-assessment, real-time feedback, simulations at any time or place without a teacher on site thanks to the possibility of sending messages during the learning process; in short, it makes online learning easier for the student and the instructor. Table 1 summarizes the differences between clinical simulation and computational simulation.

TABLE I. DIFFERENCES BETWEEN CLINICAL AND COMPUTATIONAL SIMULATIONS.

Differences		
	Clinical simulation	**Computational**
Self-assessment	No	Yes
Real-time feedback	Not recommended	Yes
Simulation at any time	No	Yes
Simulation at any place	No	Yes
Teacher needed on the premises	Yes	No
Cost	High	Lower
Major infrastructure	Yes	No
Time dedicated to each activity by the teacher (preparation of the case)	High	Low

Our contribution, in this context, consists of designing a responsive conduct model and its simulation for the creation of an educational resource with the objective of, a priori, training and in the immediate future, adding another applicability based on prediction [9]. This offers us the possibility to evaluate the possible evolutions in the shortest time attainable, according to the performances carried out by the clinician, and thus, support decision-making in the selection of the specific treatment for each patient.

In reference to current simulators, The Synergy-COPD project studies the human body's different functions through computer models at different levels (subcellular, tissue, organ and organ system) [10]. The Synergy-COPD environment is focused on preventive prediction of a disease's evolution through data obtained by the patient in a non-urgent external consultation. In contrast, our project is designed for a hospital emergency department and it allows us to observe the patient's physiological changes in relation to the interventions carried out by the healthcare staff. Authors such as Agustí et al [11] developed a theoretical multi-stage COPD computational model that dynamically integrates and graphically depicts the relationships between the exposure and inhalation of particles and gases (smoking), the biological activity (inflammatory response) of the disease, the severity of airflow limitation (FEV1) as well as the impact of the disease (dyspnea). We want to highlight the research project carried out by Reyes, A., Viciana, R., Díaz, A., and Hermida, R. [12], which is based on the design of an intelligent nucleus for a medical emergency training simulator, specifically in extra-hospital care. The reason why we want to highlight this is that our project resembles the abovementioned study [12] in its decision-making process and the changes in the patient's cardiological condition, yet our objective is focused on the emergency department and COPD patient. Without wishing to reiterate this too much, we have observed that most research projects reviewed are based on the development of models for different systems or diseases, but they always model the pathophysiological progression behavior of the disease. In contrast, our project focuses on the evolution of patients already diagnosed and exacerbated by the pathology and on checking their evolution based on the decision making carried out by healthcare staff.

This article is organised as follows: Section II shows the objectives of the research and the methodology. Section III describes the model of simulation. Section IV displays a conceptual model outline and finally, Section V closes up with a discussion and the work that is to come in the future.

II. RESEARCH OBJECTIVES AND METHODOLOGY

Chronic pathologies, especially Chronic Obstructive Pulmonary Illness (COPD), are very common in our environment.

The general objective of our research is to create a conceptual and computational model that will simulate a virtual patient with Chronic Obstructive Pulmonary Disease (COPD) in the emergency services by the use of modelling techniques.

When the patient goes to the emergency services, they are first attended by **clinicians** in a zone known as "triage", where an initial evaluation takes place. This initial evaluation consists of a few questions (their reason to be in the hospital, personal background, etc.) and parameter values (heart rate, blood pressure, etc.) so as to try to ascertain the patient's level of severity. Once the patient has been assessed, they are classified applying the triage scale used in Spain (I, II, III, IV, and V). Here they are classified in 5 different levels of acuity and priority (from I, the maximum level, to V, the minimum), following the acuity scale used in the Spanish Emergency Services, the *Andorran Triage Model*. We will focus on the patients classified as

 ISBN: 978-1-61208-756-6

level I, II and III, who will be placed in the **emergency box**. We start our investigation with patient in the Emergency Room (ER). In ER some performances will be carried out in order to know internal data about the patient (for example, monitoring their vital signs, heart rate, temperature.) Some other performances are to diagnose him (for example, requesting an analytical test) and finally, some performances will be carried out to change his evolution (for example administering a drug). In each and every one of these performances, we find variables of both qualitative and quantitative types. Due to their complexity, the variables, many of which are generated by uncertainty, will be addressed through the concept of fuzzification. Basically, this concept is a multivalued logic that allows us to mathematically represent uncertainty and vagueness, providing formal tools for their management. Table 2 shows an example:

TABLE II. SOME EXAMPLES OF VARIABLES

Variables			
Qualitative	**Value**	**Quantitative**	**Value**
Cyanosis (blue skin)	Yes/No	Temperature	Ranges (36-37,5/37,5-37,9/ 38-41)

As we can see, in the performances there are multiple variables that are to be considered but in each performance, mainly in the diagnosis (for example the analytical test) and the change of evolution (for example drug administration), time is essential in order to see the result or evolution, seeing the changes in the variables with the purpose of stabilizing the patient and discharging them.

Our methodological proposal is presented in Figure 2, bottom-up, our methodology stages will be implemented iteratively. The Iterative Model of Spiral Development will be used to develop the simulator. This system iterates permanently over the traditional development cycle of software [13]. The objective of this philosophy is to implement the model in each completed cycle so that a more complex model can be established.

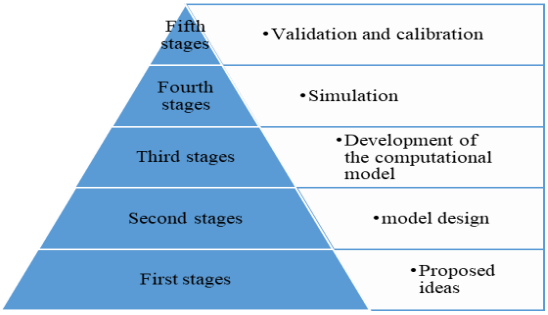


Figure 2. Stages of the suggested methodology

The long-term applicability of this investigation is, as stated previously, to support decision making, based on the prediction of the patient's evolution considering different scenarios. This will result in an important number of variables to work with and that is why we will gain from the methodology suggested, supported by the use of High-Performance Computing (HPC).

HPC enables us to run a great number of simulations, with a view to acquiring a large number of situations that in many cases would not be available without the simulation.

III. DESCRIPTION OF THE MODEL

The simulator will enhance student's training and for this it will have an interface, as shown in the example and its interpretation in Figure 3, obtained from "Pulse, The Virtual Clinical Learning Lab" [14].

The interface will feature a virtual space, as alike an emergency room as possible, where the monitoring of vital parameters, the patient and the clinical staff will be displayed. Additionally, there will be a menu, described briefly below, where the student will interact in order to obtain the data related to diagnosis and performance.

Figure 3. Example of Interface. Image obtained from Pulse. The Virtual Clinical Learning Lab.

To design the model, we propose to use Probabilistic Finite-State Machines, which is a generalization of the non-deterministic finite State Machines. It includes the probability of a given transition into the transition function. We define a finite set of states, a finite set of input, a transition function, a set of possible following states and the probability of a particular state transition taking place. To define the states, the first step has been defining the clinical variables that are to be considered and the value ranges for each variable in each situation. To change the situation, not only has the course of time been taken into consideration, but also the inputs or performances carried out by the healthcare team in the emergency service. The next step has been the creation of a state transition diagram. More details are shown below:

A. *Clinical variables*

According to the group of state variables analysed and agreed on by the Experts Committee, the following designation has been established:

- *Visibles variables*, internal variables of the system that describe the patient's state and that are known, and therefore will be shown to the student/professional (user) when they are interacting with the patient. These variables will be shown as

requested by the student, simulating the exploration, the record consulting and the tests undergone in the emergency room.

- *Not visible- monitored variables*, (internal variables of the system that describe the patient's status but that are not available to the user when they take decisions while interacting with the patient). They are taken into account in the state changes.
- *Complementary tests variables* (variables that are finally shown if requested, after a predefined period depending on the test.

B. State

The virtual patient's state is conceptually defined by the variables described above that determine the physiological time of the patient. The change of state shows changes in the variables depending on the evolution of the pathology and the decision making by the clinician and thus it will put together a new status. Each variable has a predefined set of values. The specific values that have the variables will determine the patient's specific state. Figure 4 shows an example of state:

Key
Ei=Initial status, Cy= Cyanosis yes, Ty= Tirage yes, Sy=Sleepiness yes, HR=Heart rate, FR= Respiratory rate, PosF= Fowler position, Sat= Saturation, T=Temperature, AuS=Auscultation, GasoT1= arterial blood gas type1, AnaT1=analysis T1, RxT1= X-rays type 1, HemoT1= blood culture type1.

Status 1 [(Cy), (Ty), (Sy), (PosF), (HR<100), (FR>20), (Sat<80), (T 38-41), (AuS), (GasoT1), (AnaT1), (RxT1), (HemoT1)]

Figure 4. One specific example of state

C. Inputs

We define as *Inputs* the decisions or decision making of the clinician in order to make the patient recover physiologically, involving a change in the variables that constitute the status, and turning the previous status into a different one. These so-called *Inputs* have been classified into three different types:

- *Data-attainment Inputs* are those inputs in which the student requests this information. They are related to the visible and non-visible variables mentioned before.
- *Diagnose Inputs* are those inputs that the students request to gather evidence that will help them to make a diagnosis. They are related to the complementary tests variables.
- *Performance Inputs* are those inputs that are related to the performances (drug administration, changing the patient's position, etc.), and what the student will do in order to change the patient's state.

Once the different parts that constitute the model are defined, the objective is to design a conceptual model that is summarised and explained below.

IV. CONCEPTUAL MODEL OUTLINE

Figure 5 shows an example of the interaction in which the input is implemented. This input causes some changes in the status of the conceptual model to take place, considering the time unit, and it will be shown on a monitor to the student through the interface as showed before.

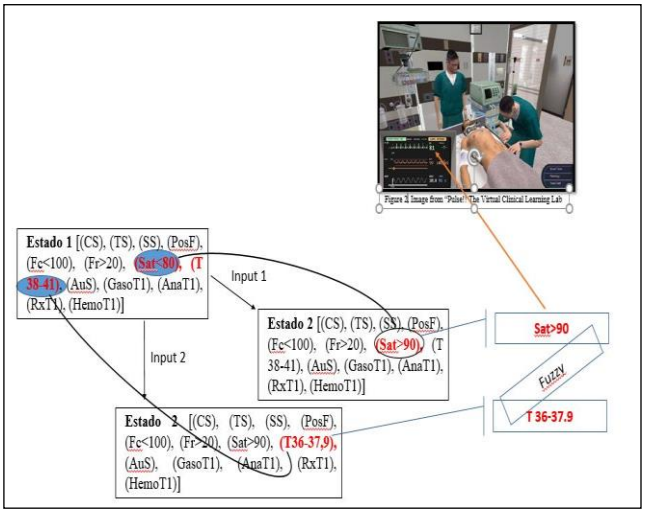


Figure 5. Example of the state changes of the conceptual model.

We propose to use Probabilistic Finite-State Machines. From each possible state that constitutes the model, each and every input will be defined and will evolve toward another state in a percentage of definite probability, considering what usually happens to the patients. Despite this, as mentioned before, the model will have a great variety of state evolution, including those that are more likely and those which are very rare in the health field but which could still happen. In relation to the latter, professors will be able to change the percentage, offering the student or professional these unusual evolutions and controlling the training to a greater or lesser extent.

Regarding some variables that constitute the state in the health field, they generate some uncertainty; there are different ranges and that means we must work with the concept fuzzy. There are different investigations in which fuzzy logic has been used by healthcare workers mainly to make decisions and, more specifically to develop models. In other words, the fuzzy logic theory combines both epistemological and philosophical vision, making it possible to understand how healthcare staff deals with complex, ambiguous and imprecise evens [15].

The verification, calibration and validation process will begin once we have the simulator. In order to calibrate and validate, we will process data on hospitalized patients from

 ISBN: 978-1-61208-756-6

two hospitals, Parc Tauli hospital and Vendrell Hospital.

V. CONCLUSION AND FUTURE WORK

Our study´s initial contribution is designing a virtual emergency room so that healthcare staff or students can develop the necessary critical thinking through decision making and make them face situations which they will face in a real environment. Simulation is a way of gaining knowledge from these kinds of situations, which can take place in real life and which on many occasions cannot be tested in a real system before they happen.

Another contribution of this study, and which is also related to knowledge, is the definition of the Method to follow in order to develop the conceptual model that will help us to develop future models of other medical conditions. The methodology suggested will be supported using High-Performance Computing. HPC allows us to make use of the simulator as a real system to carry out a great number of simulations with the aim of obtaining many scenarios; situations which in many cases are not available without the simulation.

As immediate work, and in order to make sure that the suggested model works well, it will be implemented in a computational model.

In order to do so, we will continue working in the conceptual model iteratively to end up generating a model as complex and realistic as possible.

Another long-term contribution is using such a simulator for training students and professionals; and once it has been calibrated and validated, it could be used to support the decision making based on the prediction of the patient's evolution considering of the clinical staff and thus improve the flow of patients in emergency services [16].

ACKNOWLEDGMENT

This research has been supported under contract TIN2017-84875-P, (AEI/FEDER), and partially funded by Escuelas Universitarias Gimbernat (EUG).

REFERENCES

[1] L. T. Kohn, J. M. Corrigan, and M. S. Donaldson, "To Err Is Human Building a Safer Health System," National Academy Press, 1999, pp.1-34.

[2] G. Vázquez and A. Guillamet, "Training based on simulation as an essential innovation in medical training," Educ Méd, vol. 12, num 3, 2009, pp.149–55.

[3] G. Miller, "The assessment of clinical skills /competence/performance," Acad Med, vol. 65, num 9, 1990, pp. 63-67.

[4] D. J. Scott and G. L. Dunnington, "The New ACS/APDS Skills Curriculum: Moving the Learning Curve Out of the Operating Room," J Gastrointest Surg, vol. 12, num 2, 2008 pp. 213, June. 2019, doi.org/10.1007/s11605-007-0357-y.

[5] J. Riancho, J. Maestre, I. del Moral, and J.A. Riancho, "Clinical simulation of high realism: an experience in undergraduate," Educ Méd, vol. 15, 2012, pp. 109-115.

[6] C. D. Smith, "Simulation Technology: A Strategy for Implementation in Surgical Education and Certification," Teleoperators and Virtual Environments, vol. 9, num 6, 2000, pp. 632- 637, June. 2019, doi.org/10.1162/105474600300040420

[7] S. Stansfield, D. Shawver, and A. Sobe, "Medisim: A prototype virtual reality system for training medical first responders," Sandia National Laboratories, vol. 14, num 18, 1998, pp. 198, July. 2019, doi.org/ 10.1109/VRAIS.1998.658490

[8] C. Aldrich, "Learning by doing: a comprehensive guide to simulations, computer games, and pedagogy in e-Learning and other educational experiences," San Francisco, Pfeiffer, 2005.

[9] A. Vicente, M. Antonin, D. Rexachs, and E. Luque, "Computer simulation as a methodology for theoretical Learning of clinical skills in nursing," International Journal of Integrated Care, vol. 18, num 2, 2018, pp. 297, June. 2019, doi.org/10.5334/ijic.s2297

[10] F. Miralles, D. Gomez-cabrero, M. Lluch-ariet, J. Tegnér, M. Cascante, and J. Roca, "Predictive medicine: outcomes, challenges and opportunities in the Synergy-COPD project", Journal of Translational Medicine, vol. 12, num 2, 2014, pp 1–8, September. 2019, doi.org /10.1186/1479-5876-12-S2-S12

[11] A. Alvar, A. Compte, R. Faner, J. Garcia-aymerich, G. Noell, G. Cosio, R. Rodriguez-Roisin, and J. M. Anto, "The EASI model: A first integrative computational approximation to the natural history of COPD", PLoS ONE, vol. 12, num 10, 2017, pp 1–11, September. 2019, doi.org/10.1371/journal.pone.0185502

[12] A. Reyes, R. Viciana, A. Díaz, and R. Hermida, "Training Simulator in the Field of Medical Education: Emergency Patient Modeling Based on Expert Systems", The Sixt Ibero-American Educational Informatics Congress, November 2002.

[13] B. Boehm, "A Spiral Model for Software Development and Enhacement," Computer, vol. 21, num 5, 1988, pp. 61-67, September. 2019, doi.org/10.1109/2.59

[14] The Serious Games Showcase and Challenge. "Pulse," Orland. July. 2019, http://sgschallenge.com/pulse/

[15] R. Jensen, M. Helena, and B. Moraes, "Nursing and fuzzy logic: an integrative review," Latino-Am. Enfermagem, vol. 19, num 1, 2011, pp. 195-202, September. 2019, doi.org/10.1590/S0104-11692011000100026

[16] Z. Liu, E. Cabrera, D. Rexachs, and E. Luque, "A generalized agent-based model to simulate emergency departments," The Sixth International Conference on Advances in System Simulations, IARIA, Nice, France, October 2014, pp. 65-70, July. 2019, doi.org/10.1016/j.jocs.2017.05.015

 ISBN: 978-1-61208-756-6

Simulating Strain and Motivation in Human Work Performance: An Agent-Based Modeling Approach Using the Job Demands-Resources Model

Stephanie C. Rodermund, Bernhard Neuerburg, Fabian Lorig, and Ingo J. Timm

Center for Informatics Research and Technology (CIRT)
Trier University, Behringstraße 21
54296 Trier, Germany
Email: {rodermund,neuerburg,lorigf,itimm}@uni-trier.de

***Abstract*—Even though the relevance of the "human factor" for the performance of work processes is well known, the design and optimization of such processes, e.g., in factories, often strongly focuses on machines. Especially intrinsic mental states as *strain* and *motivation* can influence the human workers' performance and thus the organizational outcome. This paper proposes an agent-based model of human work processes with respect to these intrinsic states. To this end, the well-established job demands-resources model is utilized. Experiments are presented that outline the model's capability to produce plausible results concerning the human work performance and the mutual influences between job demands, resources, and the intrinsic mental states of strain and motivation.**

Keywords–Human Work Performance; Agent-based Modeling; Job Demands-Resources Model; Strain; Motivation.

I. INTRODUCTION

Peoples' workplaces are constantly changing, especially as digitalization progresses. This digital revolution should be oriented towards employees' needs. Yet, people often subordinate to IT systems and thus disempower themselves [1]. For example, a scheduling system in a call center distributes calls without considering inidividual needs. The consequences are not only physical but also psychological strains like burn-out.

Digital change should not be rejected generally, as it has potentials for making work processes more efficient. Current approaches for designing and optimizing work processes, e.g., the production of goods in a factory, often make use of simulation and focus on machine processes such as predictive maintenance or throughput time optimization. Here, downtimes of machines or queuing strategies are analyzed to identify optimal process configurations. In reality, however, human workers can also influence the performance of such production processes, e.g., due to unavailability, distractions, or overload. Existing frameworks for the analysis of industrial service provision processes often neglect the human factor and only allow for the modeling and simulation of machines in production lines.

In a production plant, human workers may be assigned a series of orders with different difficulties to be processed during the working day. The workers' performance can be measured by the ratio of completed orders to the total number of orders. While machines usually do not show performance fluctuations when being confronted with an immense workload or time pressure, human workers are often susceptible to such influences. Intrinsic processes of motivation and strain are driving factors influencing their performance [2]. Still, human beings are often only considered as workforces without individual intrinsic needs during the planning and implementation of work processes, even though their significance and importance are well known, e.g., modeling of humans in Business Process Model and Notation (BPMN). To achieve a more adequate integration of humans into these processes as well as increase performance and organizational outcome, individuals and their intrinsic needs must be represented individually and realistically within a computer system.

This paper aims at modeling human work performance formation. Here, a special focus lies on representing the intrinsic processes of strain and motivation. For modeling workers, Agent-Based Modeling (ABM) and especially the Belief-Desire-Intention (BDI) architecture of practical reasoning [3] is proposed, as it has established in modeling of human cognitive decision-making and behaviors [4]–[7]. Therefore, in Section II, related work on the field of modeling strain and motivation in ABM is discussed. Furthermore, the flexible Job Demands-Resources model (JDR model) is introduced, which is well-established in psychology and investigates factors in the working environment that may lead to burn-out, especially focusing on those factors causing a stressful situation and mental effort for the worker [8]. Subsequently, an agent-based model of work performance is introduced in Section III. In Section IV, several experiments are conducted to analyze the model's adequacy to represent human work performance. Finally, Section V provides a summary as well as an outlook on future work.

II. BACKGROUND

There are several frameworks for modeling and optimizing work processes, e.g., Enterprise Dynamics or Anylogic [9], which focus strongly on the processes and functionalities of machines. These frameworks often neglect to represent human resources sufficiently, so that the "human factor" can not be considered properly when measuring the overall performance. However, other areas, e.g., the representation of social networks, lay emphasis on an adequate representation of the human being. Here, agent-based models that utilize sociological and psychological behavioral theories have been established [10]–[12]. This paper introduces an agent-based model of human work performance including the intrinsic processes of strain and motivation, that in future work could be used to represent workers in existing frameworks. In the

 ISBN: 978-1-61208-756-6

following, we discuss existing work on agent-based models including stress and motivation formation and present the psychological JDR model that serves as the basis for our implementation.

In ABM, various approaches exist that include psychological strain in behavioral development. Silverman's generic agent architecture contains a working memory (BDI decision logic) and four subsystems. An integrated strain value is calculated as a function of the event strain, time pressure and exhaustion, on the basis of which different coping strategies are initiated [13]. Duggirala et al. apply this conceptual model in an agent-based simulation of strain at work [14]. They selected the variables *task arrival volume*, *pending tasks*, and *work hours* to calculate the integrated strain value and to determine the coping strategies. However, by choosing work hours for exhaustion, they have missed Silverman's consideration of individual resources. Ashlock and Cage also simulate strain at work using an agent-based model and a strain factor consisting of individual strain tolerance and number of stressors [15]. Nevertheless, strain is difficult to quantify and validate, especially using static mathematical formulas that are limited to a number of variables. For this reason, Morris et al. have investigated system dynamics of strain to model agents by representing strain as causal loop diagram, and stock-flow diagram [16]. In the BDI extension BRIDGE, strain is similar to Silverman's approach part of the implicit behavior and only influences the deficiency needs and overrules selected intentions [17]. A broad field of research is crowd simulation, in whose models strain is also considered, (e.g., [18]). Strain influences behavior generation mainly reactively, but this is due to the frequent application context of emergency evacuations, where deliberative behavior is less important.

However, most models include two aspects: Firstly, the models focus on stimuli during the genesis of strain and secondly in doing so, they neglect the consideration of resources that can significantly reduce the amount of strain generated. Such models do not recognize strain as the result of intrinsic processes although psychology has already sufficiently shown the degree to which cognitive processes occur regarding strain for a long time (e.g., [19]).

In ABM regarding motivation as part of the decision-making process models can be distinguished by the motivations' directionality, i.e., whether motivation is caused by external factors or if it is merely generated intrinsically by the individual. Maslow's hierarchy of needs as an intrinsically oriented motivation theory, e.g., is implemented by Spaiser and Sumpter [20] as well as Silverman [13]. In these models, the agent's actions focus primarily on covering deficiency and growth needs, and mostly neglect environmental influences on motivation development. As already mentioned above, the BRIDGE architecture also uses this theory to define an agent's goals and desires [17]. Using Vroom's extrinsically oriented expectation theory, the agent's decision making is modeled on the basis of its expected subjective value of a future event in his environment [21], [22]. These models mainly rely on subjectively perceived environmental factors and largely neglect the mutual influence of intrinsic factors, e.g., between perceived strain and motivation, although the relation between these factors has already been recognised (see, e.g., [17]).

A well-known model that both considers stressors (stimuli), resources, and the influence of motivation, is the JDR model by Demerouti et al. [8]. The JDR model is an empirically evaluated model, that has been used flexibly in a variety of scenarios, such as to predict job burn-out [23], organizational commitment [24], connectedness [25], and work engagement [26]. The model consists of two essential processes: a health impairment process and a motivational process (see Figure 1). The health impairment process is concerned with how job demands affect individual strain. Job demands can be stressors like workload, emotional demands, or organizational changes.

During the motivational process, job resources are main predictors for motivation and engagement. Whereas job demands consume energetic resources and in this way cause strain, job resources fulfil basic psychological needs and therefore generate motivation. Thus, job demands and resources initiate two different processes, but these processes are not independent, because job resources can buffer the impact of job demands on strain and job demands can reduce the generation of motivation through job resources (see Figure 1). Due to these moderation effects, there is also a direct relationship between strain and motivation. By using the model, predictions can be made about employee well-being, job-performance, and respectively the aggregated performance of a company. The model was extended several times by the authors, in particular to include personal resources and job crafting, and was maturated into a theory based on several meta-analyses [2]. Nevertheless, this paper uses the original model to reduce the complexity of the simulation significantly and focus on the prediction of job performance [28].

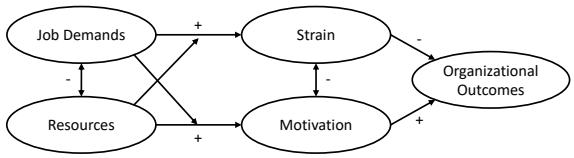


Figure 1. Job Demands-Resources Model [27].

III. An Agent-Based Model of Work Performance

In this section, an agent-based model of human work performance is introduced that coombines the BDI architecture and the JDR model presented in Section II. The workers are modeled based on the BDI architecture of practical reasoning [3]. It organizes goals (desires), information about the environment and the own conditions (beliefs), and action-oriented measures (intentions) into mental states. To this end, we make use of the JDR model presented in Section II. By utilizing both models a strict modularization is made, which can be easily extended and exchanged against further theories and models.

Figure 2 shows the basic concept of the agent-based model of human work performance. Following the JDR model, the agent's environment consists of sets of $JobDemands$ and $JobResources$, that impact internal processes forming $strain$ (α) and $motivation$ (β). These, in turn, determine the agent's action and thus the organizational outcomes (here: equal to individual performance).

Referring to the factory example introduced in Section I, the agent is confronted with a set of $Orders$, that are composed of the two sets $UnfinishedOrders$ and $FinishedOrders$ (Equation 1). Initially, $|Orders|$ is equal to $|UnfinishedOrders|$. If

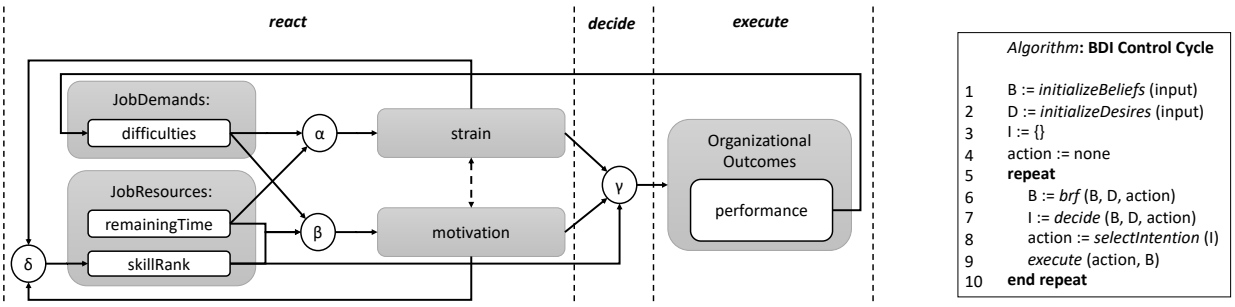


Figure 2. Concept of Job demands-Resources Model in Agent-Based Model (left) and Algorithm (right).

an order $i \in UnfinishedOrders$ is completed, it is deleted from this set and added to $FinishedOrders$. Each of the orders has a certain difficulty $diff_i \in \mathbb{N}$, which is defined within a range of set difficulties. The difficulty of an order expresses how much time is required to execute the order. Because job demands represent stressors as workload (see Section II) the variable $difficulties$ is introduced, which represents the agent's workload on one working dayand is composed of the sum of difficulties $diff_i$ for each $i \in UnfinishedOrders$ (Equation 2).

$$Orders = FinishedOrders \bigcup UnfinishedOrders \quad (1)$$

$$difficulties = \sum_{i=1}^{|UnfinishedOrders|} diff_i \quad (2)$$

One working day is defined by a number of time steps $totalTime \in \mathbb{N}$, whereas the running variable $t \in \mathbb{N}$ represents the current time that has already elapsed. At each time step, the $remainingTime$ to complete all $UnfinishedOrders$ is computed (Equation 3). The difficulty level corresponds to the minimum number of time units required to process an order and depends on the agent's $skillRank \in \mathbb{N}$, thus its work-related know-how. A lower value of $skillRank$ means, that less time units are needed to complete one difficulty level. The $skillRank$ together with the overall $remainingTime$ to complete all orders form the agent's set of $JobResources$.

$$remainingTime = totalTime - t \quad (3)$$

Considering the JDR model, job demands initate a health impairment process that affects the agent's individual strain. Job resources on the other hand, have a moderating effect on strain and buffer the impact of the job demands. Therefore, $strain$ (Figure 2, Function α) represents the experienced pressure as the ratio between the unfinished orders $difficulties$ and the $remainingTime$ to complete them (Equation 4). $Motivation$ is formed in a process that is mainly influenced by job resources. Here, job demands reduce the generation of this variable. Hence, $motivation$ (Figure 2, Function β) implies the capabilities of the agent and represents whether the agent is able to perform the open orders in the given time based on its own $skillRank$ at time t (see Equation 5). If the $motivation$ value is high, the agent is confident to complete the whole set of unfinished orders in the remaining time. $Strain$ and $motivation$ represent the agent's set of $IntrinsicStates$. Both values are normalized to $[0, 1]$ relative to the minimal and maximal possible values of the variables.

$$strain = \frac{difficulties}{remainingTime} \quad (4)$$

$$motivation = \frac{remainingTime}{skillRank_t * difficulties} \quad (5)$$

Following the example introduced in Section I, $performance$ is measured using the ratio of $FinishedOrders$ to the overall number of $Orders$ (Equation 6).

$$performance = \frac{|FinishedOrders|}{|Orders|} \quad (6)$$

The algorithm in Figure 2 shows the BDI control cycle, that determines the agent's behavior formation process. First, the internal states, as well as a variable determining the next action to perform are initally set (lines 1-4). Based on the general BDI architecture, the agent's behavior in our model is formed by passing various phases that consider and build the mental states. These can be divided into *react*, *decide*, and *execute* (see [29]). In *react* (*belief-revision-function (brf)*), the agent processes perceived information and updates its beliefs (B) about the current situation and intrinsic states. In *decide*, based on the updated beliefs and the agent's desires (D), the agent updates its intentions (I). Considering these, an action to perform next is chosen, before it is carried out in *execute*.

The agent's beliefs B are composed of the three sets $JobDemands$, $JobResources$, and $IntrinsicStates$ (see Equation 7). Based on the beliefs B that are generated and updated in *react*, the agent decides for an unfinished order to proceed with next, to reach its sole desire (completing all orders).

$$B = JobDemands \bigcup JobResources \bigcup IntrinsicStates$$
$$\Rightarrow B = \{difficulties, remainingTime, skillRank, strain, motivation\} \quad (7)$$

In the *decide* phase, best choices for both values of $strain$ and $motivation$ are defined. Hereby, a mapping of $UnfinishedOrders$' difficulties $diff_i$ to the respective values takes place, whereas a high $motivation$ value leads to a choice of a high difficulty. A high value of $strain$ generates a low difficulty as the best possible choice. These best choices serve as boundaries to decide for the intention I to commit to, which is done on a random basis (see Figure 2, Function γ). Consequently, *decide* is only processed if the current order has been completed in the preceding time step. The

chosen difficulty (I) is used to pick the next order ($action$) to complete, which is then performed in *execute*. Starting from the initial value the $skillRank$ adapts in dependence to the values of $motivation$ and $strain$ (decrease or increase of value) and to the current order's difficulty (strength of decrease or increase of value) (Figure 2, Function δ). After each time step t, the $performance$ is used to update the orders' difficulties.

IV. Simulating Work Performance: Experiments and Results

In this section, the agent-based model of work performance is evaluated based on a case study. Therefore, first the simulation setup is defined and the model input variables are specified. Subsequently, the findings are presented and the assumptions derived from these are discussed.

A. Simulation Setup

In order to simulate human work performance, the variables introduced in Section III are specified. The number of orders $|Orders|$ is set to 20 and one agent is simulated at a time. Because it is theorized that the workers abilities do not vanish completely due to an immense workload, the $skillRank$ cannot exceed a maximum of 10. Equally, the skills of an agent can not decrease infinitely and the maximum decrease of the $skillRank$ is to 1, so that at least one time unit is needed to complete one difficulty level of an order. Each ten time units, the values of $strain$, $motivation$, and $skillRank$ are updated.

To test the impacts of input variables on performance, 27 scenarios are defined and the output behavior of the model is analyzed (see Figure 3). Each scenario is defined by an element of the cartesian product of $timeCapacity$, $difficultyRange$, and $skillRank$ (see Table I). The agent's inital $skillRank$ is varied between a minimal (1), medium (5), and maximum value (10). The orders' $difficultyRange$ encompasses the range 1-5 (complete difficulty range), 1-3 (low difficulty), and 3-5 (high difficulty).

TABLE I.
SCENARIO SPECIFICATION.

timeCapacity		*difficultyRange*		*skillRank*
smallTimeCapacity		1-3		1
suitableTimeCapacity	×	1-5	×	5
highTimeCapacity		3-5		10

$TimeCapacity$ is defined by a $small$, $suitable$, and $high\ timeCapacity$ and depicts the time available to perform the set of $Orders$. The three values of $small$ (405) and $high\ timeCapacity$ (135) to complete the orders center around a suitable amount of time units (270). The reference value of 270 time units arises from the time an agent persistently having a $skillRank$ of 5 would need to complete all orders with difficulty range from 1-5. Because the model makes use of random number generators, it is repeated 30 times to neglect possible effects.

B. Simulation Results and Discussion

The diagrams of Figure 3 show the experimental results seperated by the variation of $timeCapacity$. The x-axis depicts the initial input value of the variable $skillRank$. The y-axis shows the performance of the agent, thus, the ratio of finished orders compared to the overall number of orders. The boxplots' colors represent the orders' $difficultyRange$, darkgrey represents a range of 1-3, lightgrey for 1-5, and white for a range of 3-5. In the following, three main observations regarding the work performance are discussed.

First, the results show that with increasing $timeCapacity$, the agent is able to finish all or a majority of orders. Where $small\ timeCapacity$ shows a minimum performance value of 0.2, this value increases to 0.55 in $high\ timeCapacity$, so that even an agent with a $skillRank$ of 10 is capable of completing half of the orders. Theoretically, an agent with $skillRank$ of 1 should be able to complete the whole set of orders even in the $small\ timeCapacity$ scenarios. Nevertheless, this mainly occurs in a $high\ timeCapacity$ scenario. The simulation output does not reveal the difficulties of the finished orders. Agents with a small $skillRank$ tend to choose orders with a high difficulty due to a high $motivation$ value. Thus, the agent may finish these high difficulty orders but has a worse performance due to an overall smaller number of finished orders. Agents with a $skillRank$ of 10, on the other hand, tend to start with low difficulty orders which is why the number of finished orders differs less than expected from that of the remaining $skillRanks$. With increasing time capacity the choosing behavior becomes less important, because the agent still has enough time left to finish the remaining orders of small difficulty.

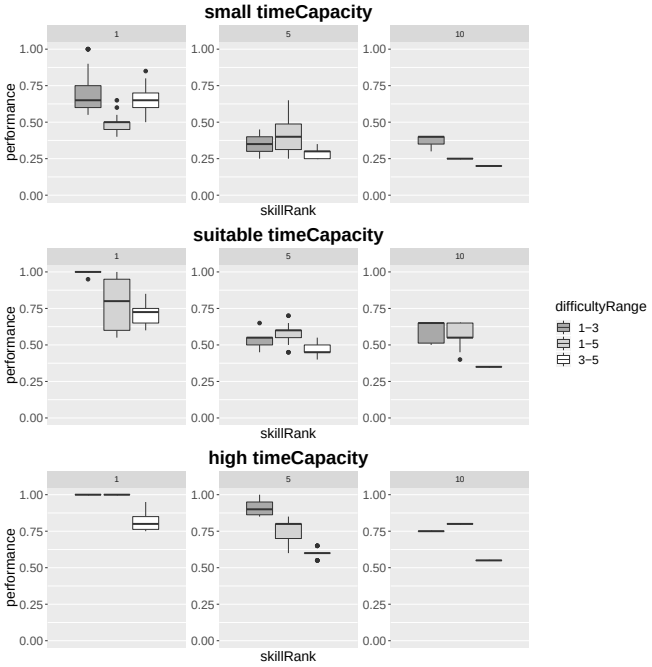


Figure 3. Performance depending on $timeCapacity$, $skillRank$, and $difficultyRange$.

The second trend observed is the influence of the $difficultyRange$ of orders on performance. The results indicate that a range of 3-5 leads to the worst performance. Thus, the performance mean throughout the simulation runs is 0.52, whereas ranges 1-3 and 1-5 lead to mean values of 0.69 and 0.63. Transferred to working life, a constant execution of difficult orders that afford a high level of exertion and concentration may lead to a low performance due to a heavy

strain load. A balanced order compilation is more purposeful as it, on the one hand, demands the worker enough to keep his interest, and on the other hand, allows for phases of lower concentration while completing orders of a low difficulty level [30].

Nevertheless, there are exceptions to this observation, as it can be observed in $small\ timeCapacity$ with $skillRank$ of 1 and 5 as well as in the suitable time capacity scenario with $skillRanks$ 5 and 10. A first exception is a worse performance for order difficulty range 1-5 than for 3-5 ($small\ timeCapacity$ and $skillRank = 1$). This can be traced back to the fact that the agent starts the simulation with a low level of $strain$ as well as a high $motivation$ value, due to the wider range of 1-5 and thus a smaller value of $difficulties$. In contrast, a second exception intimates a worse performance for range 1-3 than for 1-5 ($small\ timeCapacity$ and $skillRank = 5$; $suitable\ timeCapacity$ and $skillRank = 5$ or 10). The agents in these scenarios start with a relatively low $motivation$ due to the comparably high value of $skillRank$. However, the $difficulties$ absolutely define which order is chosen first. Therefore, in both exceptions, the agent chooses high difficulties first which, caused by the progressing time, leads to increasing $strain$ and decreasing $motivation$ and ultimately to less finished orders. The range of order difficulties should thus be adapted to the available time as well as the agent's skills in order to reach the best results.

A third tendency refers to the influence of the input $skillRank$ on performance. In contrary to the remaining values, a $skillRank$ of 10 tends to lead to extreme performance values without any outliers. Especially concering $difficultyRange$ 3-5, the agent is not capable of completing more than 20% of the existing orders. This is due to a low $motivation$ value resulting from the high $skillRank$ as well as the restriction of the model to generate a higher $skillRank$ than 10. With decreasing $remainingTime$, the $strain$ value increases and the $skillRank$ is not allowed to improve. This covers findings in psychology that investigate the connection between high strain at the workplace to burn-out and thus a low job performance [31].

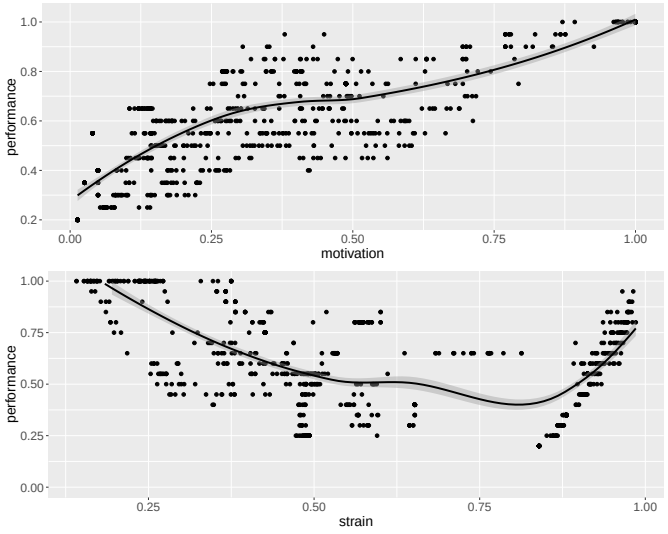


Figure 4. Performance depending on $motivation$ and $strain$.

The diagrams in Figure 4 depict the impact of the values of $motivation$ and $strain$ (x-axis) on the agent's performance (y-axis). The data points represent the mean value of the variables $motivation$ and $strain$ throughout one simulation run (x-axis), whereas performance refers to the status at the end of a simulation run (y-axis). The diagrams undermine the aforementioned tendecies. With increasing mean $motivation$ throughout the simulation runs, the agent's performance increases almost lineary. Each scenario with a mean $motivation$ value of 1.0 leads to a perfect performance, whereas a value $<= 0.25$ results in a mean performance of 0.42 and a best value of 0.75. The agent's $strain$ value shows a u-shaped impact on performance. A low $strain$ level ($<= 0.25$) leads to a high performance mean of 0.99. Subsequently, the curve drops to a minimal mean performance of 0.2 at a $strain$ level of 0.84, before it starts to increase again to a mean value of 0.86 at a $strain$ level of 0.98. The upswing of the curve at higher $strain$ values is rather unexpected, since there is a linear calculation of the variable. Once again, the diagram does not show us the difficulties of unfinished orders, but the proportion of finished orders in comparison to all orders. Hence, if the agent is heavily strained during the entire simulation run, its $skillRank$ probably rises fast to the highest possible value and remains there for the rest of the run. This causes the agent to favor low difficulty orders, which increases the value of $motivation$ and performance.

V. Conclusion and Future Work

In this paper, an agent-based model of human work performance was presented that makes use of the JDR model. A decision-behavior based on the general BDI architecture was introduced and adapted to the processes defined in the JDR model including a representation of strain and motivation as well as the mutual influences of job resources, job demands, and intrinsic mental states. Within several experiments, the impacts of the input variables $timeCapacity$, $skillRank$, and $difficultyRange$ on the overall performance of the agents were analyzed. The experimental results revealed that the model is capable of producing realistic working performance including intrinsic processes of strain and motivation. However, the model lacks reliable empirical studies to validate the model and the underlying relationships.

In future work, we plan on conducting empirical experiments with workers in a controlled working environment (see, e.g., [32]). During these experiments, we aim at identifying stressors and resources and measure individual reactions like strain, especially by biosignals. Furthermore, we need to improve the existing model in several respects. As mentioned in Sec. IV-A, a $skillRank$ of 10 leads to performance measures that indicate burn-out developing processes. In order to investigate at which point the agent is no longer able to perform, further experiments with more detailed parameter steps have to be executed. Hereby, possible intervention strategies to countermeasure this development could be tested. The model shows the best results for orders within difficulty range 1-3. As has been discussed in Sec. IV-A, a varied order difficulty should lead to best performances, due to a balanced ratio of exertion and relaxation. In order to receive a more realistic representation, the effects of missing challenges could be included. A difficulty range of 1-3 would thus theoretically lead to a worse performance than a range of 1-5. The agents'

 ISBN: 978-1-61208-756-6

performance should be measured by showing how much of the workload has been completed. Thus, not only the proportion of finished orders, but the difficulties of the finished orders should be taken into account, too. Furthermore, working in teams should be included in the model. This could result in better organizational outcomes, as by the interaction, poor performances of some members may be offset by good performances of others.

REFERENCES

[1] H. Kagermann, "Chancen von Industrie 4.0 nutzen (Seizing opportunities of Industry 4.0)," in Industrie 4.0 in Produktion, Automatisierung und Logistik: Anwendung · Technologien · Migration, T. Bauernhansl, M. ten Hompel, and B. Vogel-Heuser, Eds. Wiesbaden: Springer Fachmedien Wiesbaden, 2014, pp. 603–614.

[2] A. B. Bakker and E. Demerouti, "Job demands–resources theory," Wellbeing: A complete reference guide, 2014, pp. 1–28.

[3] A. S. Rao and M. P. Georgeff, "BDI agents: from theory to practice." in ICMAS, vol. 95, 1995, pp. 312–319.

[4] E. Bonabeau, "Agent-based modeling: Methods and techniques for simulating human systems," Proceedings of the National Academy of Sciences, vol. 99, no. Supplement 3, May 2002, pp. 7280–7287.

[5] W. Jager and M. Janssen, "The Need for and Development of Behaviourally Realistic Agents," in Multi-Agent-Based Simulation II, G. Goos, Hartmanis, J., van Leeuwen, J., S. Sichman, J., F. Bousquet, and P. Davidsson, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, vol. 2581, pp. 36–49.

[6] J. O. Berndt, S. C. Rodermund, and I. J. Timm, "Social Contagion of Fertility: An Agent-based Simulation Study," in Proceedings of the 2018 Winter Simulation Conference (WSC). Gothenburg, Sweden: IEEE, Dec. 2018, pp. 953–964.

[7] L. Reuter, J. Berndt, and I. Timm, "Simulating Psychological Experiments: An Agent-Based Modeling Approach," in Proceedings of the Fourth International Conference on Human and Social Analytics (HUSO 2018), Wilmington, DE, USA, 2018, pp. 5–10.

[8] E. Demerouti, A. Bakker, F. Nachreiner, and W. Schaufeli, "The job demands-resources model of burnout." Journal of Applied Psychology, vol. 86, no. 3, 2001, pp. 499–512.

[9] E. Serova, "The role of agent based modelling in the design of management decision processes," The electronic journal information systems evaluation, vol. 16, no. 1, 2013, pp. 71–80.

[10] M. W. Macy and R. Willer, "From factors to actors: Computational sociology and agent-based modeling," Annual review of sociology, vol. 28, no. 1, 2002, pp. 143–166.

[11] T. Balke and N. Gilbert, "How do agents make decisions? a survey," Journal of Artificial Societies and Social Simulation, vol. 17, no. 4, 2014, pp. 13–.

[12] E. R. Smith and F. R. Conrey, "Agent-based modeling: A new approach for theory building in social psychology," Personality and social psychology review, vol. 11, no. 1, 2007, pp. 87–104.

[13] B. G. Silverman, "More Realistic Human Behavior Models for Agents in Virtual Worlds: Emotion, Stress, and Value Ontologies," University of Pennsylvania/ACASA Technical Report, Tech. Rep., 2001.

[14] M. Duggirala, M. Singh, H. Hayatnagarkar, S. Patel, and V. Balaraman, Understanding Impact of Stress on Workplace Outcomes Using an Agent Based Simulation. Unpublished, 2016.

[15] D. Ashlock and M. Page, "An agent based model of stress in the workplace," in 2013 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS). IEEE, 2013, pp. 114–121.

[16] A. Morris, W. Ross, and M. Ulieru, "A system dynamics view of stress: Towards human-factor modeling with computer agents," in 2010 IEEE International Conference on Systems, Man and Cybernetics. IEEE, 2010, pp. 4369–4374.

[17] F. Dignum, V. Dignum, and C. M. Jonker, "Towards agents for policy making," in International Workshop on Multi-Agent Systems and Agent-Based Simulation. Springer, 2008, pp. 141–153.

[18] Y. Mao, S. Yang, Z. Li, and Y. Li, "Personality trait and group emotion contagion based crowd simulation for emergency evacuation," Multimedia Tools and Applications, 2018, pp. 1–28.

[19] R. S. Lazarus and S. Folkman, Stress, appraisal, and coping. Springer publishing company, 1984.

[20] V. Spaiser and D. J. T. Sumpter, "Revising the Human Development Sequence Theory Using an Agent-Based Approach and Data," Journal of Artificial Societies and Social Simulation, vol. 19, no. 3, 2016.

[21] A. Sharpanskykh, "Modeling of Agents in Organizational Context," in Multi-Agent Systems and Applications V, H.-D. Burkhard, G. Lindemann, R. Verbrugge, and L. Z. Varga, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, vol. 4696, pp. 193–203.

[22] A. Sharpanskykh and S. H. Stroeve, "An agent-based approach for structured modeling, analysis and improvement of safety culture," Computational and Mathematical Organization Theory, vol. 17, no. 1, Mar. 2011, pp. 77–117.

[23] A. B. Bakker, E. Demerouti, and M. F. Dollard, "How job demands affect partners' experience of exhaustion: Integrating work-family conflict and crossover theory." Journal of Applied Psychology, vol. 93, no. 4, 2008, pp. 901–911.

[24] A. B. Bakker, M. Van Veldhoven, and D. Xanthopoulou, "Beyond the demand-control model: Thriving on high job demands and resources," Journal of Personnel Psychology, vol. 9, no. 1, 2010, pp. 3–16.

[25] K. A. Lewig, D. Xanthopoulou, A. B. Bakker, M. F. Dollard, and J. C. Metzer, "Burnout and connectedness among Australian volunteers: A test of the Job Demands–Resources model," Journal of vocational behavior, vol. 71, no. 3, 2007, pp. 429–445.

[26] A. B. Bakker, J. J. Hakanen, E. Demerouti, and D. Xanthopoulou, "Job resources boost work engagement, particularly when job demands are high." Journal of educational psychology, vol. 99, no. 2, 2007, pp. 274–284.

[27] A. B. Bakker and E. Demerouti, "The job demands-resources model: State of the art," Journal of managerial psychology, vol. 22, no. 3, 2007, pp. 309–328.

[28] D. Xanthopoulou, A. B. Bakker, E. Demerouti, and W. B. Schaufeli, "The role of personal resources in the job demands-resources model." International journal of stress management, vol. 14, no. 2, 2007, pp. 121–141.

[29] I. J. Timm, Dynamisches Konfliktmanagement als Verhaltenssteuerung Intelligenter Agenten (Dynamic conflict management as behavior control of intelligent agents), ser. Dissertationen zur Künstlichen Intelligenz (DISKI). Berlin: Akad. Verl.-Ges. Aka, 2004, no. 283.

[30] W. B. Schaufeli and M. Salanova, "Burnout, boredom and engagement in the workplace," in People at work: An introduction to contemporary work psychology. New York, NY: Wiley, 2014, pp. 293–320.

[31] C. Maslach, W. B. Schaufeli, and M. P. Leiter, "Job Burnout," Annual Review of Psychology, vol. 52, no. 1, Feb. 2001, pp. 397–422.

[32] A. Eckhardt, C. Maier, and R. Buettner, "The influence of pressure to perform and experience on changing perceptions and user performance: A multi-method experimental analysis," in Proceedings of the International Conference on Information Systems., 2012.

 ISBN: 978-1-61208-756-6

On the Calibration, Verification and Validation of an Agent-Based Model of the HPC Input/Output System

Diego Encinas, Marcelo Naiouf, Armando De Giusti

Informatics Research Institute LIDI
CIC's Associated Research Center
Universidad Nacional de La Plata
50 y 120, La Plata, 1900, Argentina
Email: {dencinas, mnaiouf, degiusti}@lidi.info.unlp.edu.ar

Sandra Mendez

Computer Sciences Department
Barcelona Supercomputing Center (BSC)
Barcelona, 08034, Spain
Email: sandra.mendez@bsc.es

Dolores Rexachs and Emilio Luque

Computer Architecture and Operating Systems Department
Universitat Autònoma de Barcelona
Bellaterra, 08193, Spain
Email: {dolores.rexachs, emilio.luque}@uab.es

***Abstract*—High Performance Computing (HPC) applications can spend a significant portion of their execution time making Input/Output (I/O) operations into files. Improving I/O performance becomes more important for the HPC community as parallel applications produce more data and use more compute resources. One of the methods used to evaluate and understand the I/O performance behavior of such applications in new I/O system or different configurations is using modeling and simulation techniques. In this paper, we present a simulation model of the HPC I/O system by using Agent-Based Modelling and Simulation (ABMS) based on the functionality of the I/O Software Stack. Our proposal is modeled using the concept of white box so that the specific behavior of each of the modules or layers in the system can be observed. The I/O software stack layers are analyzed using code instrumentation for the features corresponding to I/O operations and calibration of the initial model. The verification and validation of an initial implementation has shown a similar behavior between the measured and simulated values for the Interleaved or Random (IOR) benchmark by using different file sizes.**

Keywords–Agent-Based Modelling and Simulation (ABMS); HPC-I/O System; Parallel File System.

I. INTRODUCTION

Many scientific applications benefit considerably from the rapid advance of processor architectures used in the modern High Performance Computing (HPC) systems. However, they can spend a significant portion of their execution time making Input/Output (I/O) operations into files. Inefficient I/O is one of the main bottleneck for scientific applications in a large-scale HPC environment.

In the HPC field, the I/O strategy recommended is the parallel I/O that is a technique used to access data in one or more storage devices simultaneously from different application processes so as to maximize bandwidth and speed up operations. For its implementation, a parallel file system is required; otherwise the file system would probably process the I/O requests it receives sequentially, and no specific advantages in relation to parallel I/O would be gained.

Generally, evaluating the performance offered by a HPC I/O system with different configurations and the same application allows selecting the best settings. However, analyzing application performance can also be a useful before configuring the hardware.

One of the methods used to predict different application configurations behavior in a computer system is using modeling and simulation techniques. That is, analyzing and designing simulation models based on the parallel I/O architecture allows reducing complexity and fulfilling application requirements in HPC by identifying and evaluating the factors that affect performance.

There are several research efforts in HPC I/O system simulators focusing on storage architecture and some layers of the I/O software stack. The Simulator Framework for Computer Architectures and Storage Networks (SIMCAN)[1] is oriented to optimizing communications and I/O algorithms. The Parallel I/O Simulator of Hierarchical Data (PIOSimHD) [2] was developed to analyze Message Passing Interface-Input/Output (MPI-I/O) performance. The Co-design of Exascale Storage System (CODES) [3] is a framework developed to evaluate the design of the exascale storage systems. The High-Performance Simulator for Hybrid Parallel I/O and Storage System (HPIS3) [4] models application workload.

CODES and HPIS3 are based on Rensselaer's Optimistic Simulation System (ROSS) [5], which is a parallel simulation platform. SIMCAN was developed using OMNET++, and PIOSimHD was programmed in Java. All the tools mentioned use an event-based simulation paradigm (Discrete Event Simulation, DES). We propose to develop a simulator using Agent-Based Modeling and Simulation (ABMS) that will allow evaluating the performance of the I/O software stack. The agent paradigm is used in various scientific fields and is of special interest in Artificial Intelligence (AI), it allows successfully solving complex problems compared with other classic techniques [6]. It is a simulation technique that recreates the functionality of different components in a real system by modeling entities known as agents. Basically, an agent is an entity capable of perceiving and acting based on changes in its environment. It can also interact with other agents, executing and coordinating its actions, to achieve goals.

 ISBN: 978-1-61208-756-6

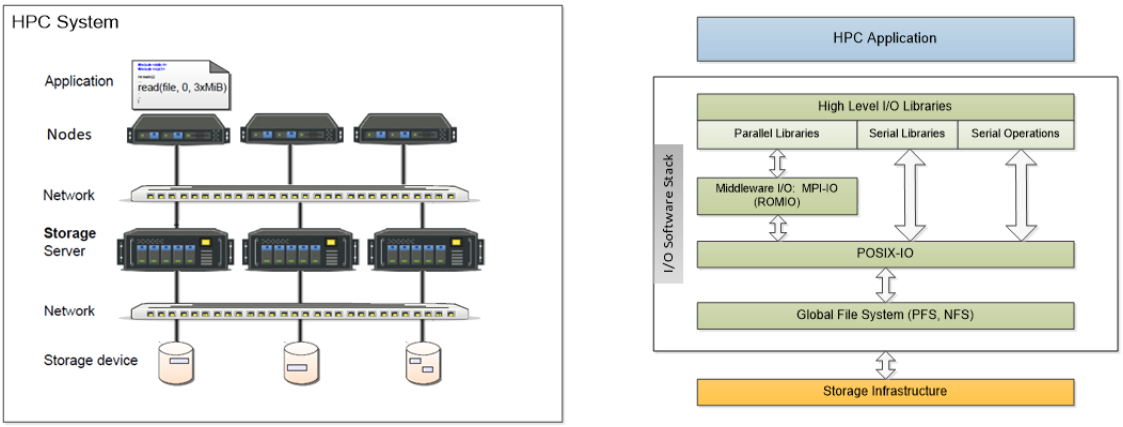


Figure 1: A typical HPC System and the I/O Software Stack

Generally, both paradigms operate in discrete time, but DES is used for low to medium abstraction levels. In ABMS, system behavior is defined at an individual level, and global emergent behavior appears when the communication and interaction activities among the agents in an environment start. In fact, ABMS is easier to modify, since model debugging is usually done locally rather than globally [7].

An advantage of ABMS is that different types of models could be created for each part of the system [8][9]. This is useful since the behaviors of the models differ from each other as they are related to diverse actions like processing, communications and storage. Furthermore, the environments could be both software and hardware. ABMS allows it to implement different components in a modular and flexible way, affording the possibility of connecting and disconnecting different parts of a complex system for a layer-level analysis.

In this paper, we present a model of the HPC-IO system for an initial simulation using ABMS. Our proposal is modeled using the concept of white box so that the specific behavior of each of the modules or layers in the system can be observed. The I/O software stack layers were analyzed using code instrumentation for the features corresponding to I/O operations and calibration of the initial model. The verification and validation of the proposed model has shown similar I/O behavior on the real and simulated environment.

The rest of this paper is organized as follows. Section II briefly describes I/O system in HPC. Section III addresses a functionality analysis for the development of the initial conceptual model. Section IV describes the proposed model, initial implementation and validation. Finally, Section V presents our conclusions and future works.

II. BACKGROUND

The I/O subsystem in the HPC area consists of two abstraction levels, software and hardware. Usually, the I/O Software includes parallel file system and high level I/O libraries and the I/O hardware refers to storage devices and networks. However modern HPC I/O system can include more components increasing the complexity of the I/O system.

Figure 1 illustrates the structure of the I/O software stack. An I/O operation goes through the software stack from the user application up until it obtains access to the disk from where data are read or on which data are written. Since this parallelism is complex to coordinate and optimize, the implementation of intermediate several layers was designed as a solution.

A. HPC I/O Strategies

The most common I/O strategies in HPC are the serial or parallel accesses into files. Serial I/O is carried out by a single process and it is a non-scalable method because operation time grows linearly with the volume of data and even more with the number of processes, since more time will be required to collect all data in a single process [10].

Parallel I/O usually presents two methods or variations of them: ”One file per process” and ”a single shared file”. In ”one file per process”, each process reads/writes data on its own file on disk and no coordination is required among processes. ”One single shared file” is more convenient to implement Parallel I/O, where all processes write to the same file on disk, but on different sections of that file. This method requires a shared file system that is accessible to all processes.

There are two ways in which multiple processes can access a shared file: independent access and collective access. In the first case, each process accesses the data directly from the file system without communicating or coordinating with the other processes. In collective access, small and fragmented accesses are combined into larger ones to the file system that helps significantly reduce access times. Our aim is to identify this kind of optimizations to explain the I/O behavior, for this reason, we propose a white box model.

B. Middleware

MPI is an interface and communications protocol used to program applications in parallel computers. It is designed to provide basic virtual topology, synchronization, and communication functionalities within a set of processes in an abstract way that is independent from the programming language used to develop the application.

MPI-IO functions work in similar way to those of MPI: writing MPI files is similar to sending MPI messages, and reading MPI files is like receiving MPI messages. MPI-IO also allows reading and writing files in a normal (blocking) mode, as well as asynchronously, to allow performing computation operations while the file on storage device is being read or written on the background. It also supports the concept of collective operations: each process can access MPI files on its own or all together, simultaneously. The second alternative

 ISBN: 978-1-61208-756-6

offers greater reading and writing optimizations that can be implemented on several levels. Mostly of MPI distribution provides MPI-IO functions by using ROMIO [11], which is an implementation of MPI-IO standard and it is used in MPI distributions, such as MPICH, MVAPICH, IBM PE and Intel MPI.

C. Parallel File Systems

A parallel file system is a distributed file system that stripes the files data into multiple data servers, connected to storage devices that provide concurrent access to the files through multiple tasks of a parallel application run on a cluster. The main advantages offered by a parallel file system include a global name space, scalability, and the ability to distribute large files through multiple storage nodes in a cluster environment, which makes a file system like this very appropriate for I/O subsystems in HPC. Typically, a parallel file system includes a metadata server with information about the data found on the data servers.

Some systems use a specific server for metadata, while others distribute the functionality of a metadata server through the data servers. Some examples of parallel file systems for high performance computing clusters are IBM Spectrum Scale, Lustre and PVFS2.

PVFS offers three interfaces through which PVFS files are accessed: PVFS' native Application Programming Interface (API), Linux kernel's interface, and ROMIO interface. The latter uses MPI to access PVFS files through MPI-IO's interface.

The underlying complexity of sending requests to all storage nodes and sorting file contents, among other tasks, is handled by PVFS. When a program attempts a reading operation on a file, small sections of the file are read from several storage devices in parallel. This reduces the load on any given disk controller and allows handling a larger number of requests.

D. Benchmarks

To evaluate the performance of parallel file system and test different I/O libraries of the I/O software stack, exists different I/O benchmark. Benchmarks are designed to mimic a specific type of workload in a component or system. One the most accepted I/O benchmark in HPC is IOR[12]. It supports several application I/O patterns and allows configuring them, and it offers access to shared files both independently and collectively. Additionally, IOR offers different execution options for the same algorithm using various parallel programming interfaces, including POSIX, MPI-IO, HDF5 and NETCDF.

III. FUNCTIONALITY ANALYSIS

To define an initial model of the I/O system, system functionality should be fully understood. First, the I/O pattern type to be analyzed was selected, and then the corresponding software stack layers for this model were applied. We have selected the IOR benchmark to evaluate I/O performance in HPC clusters. The analysis focused on the functionality that was observed for IOR in the data path.

Due to the heterogeneity of the I/O systems and the complexity of the software stack, an analysis was started for MPI-IO layers and the parallel file system. PVFS2 was the file system selected for our tests. At this time, we separated the different components considering the concepts of a parallel file system to allow us using the model with other parallel file systems, such as Lustre in the future.

The IOR benchmark offers the total runtime measurements for their programs, but they do not go into further detail in relation to the different abstraction layers of the parallel I/O system. These layers have to be crossed from the moment the user application sends an I/O request up until the CPU, through its operating system, effectively accesses the file on disk to read or write the data. Therefore, it is important to identify the layer in the software stack that requires more time during an I/O operation.

To follow the data path in the software stack, tracers or monitors can be used, but these operate on different levels of the I/O system. There is no single tool that allows recording the I/O behavior in all levels. For this reason, code instrumentation has been implemented in both the MPI-IO layer and the parallel file system layer.

A. Code Instrumentation

One possible way for finding out how the different modules in the I/O request process (application, middleware and file systems) work is instrumenting each of them by adding small sections of code. Thus, it would be possible to establish what percentage of the total runtime of an I/O operation corresponds to each of them, which would help knowing which of them is the most critical one and should be enhanced to dramatically improve parallel I/O speed.

Additional source code sections are simply a few lines of code written in C to monitor and measure runtime for some functions that were identified as critical during the request, service, and execution process of an I/O operation as it goes through each of the abstraction levels of the parallel system. To avoid hindering or interfering with the benchmark result screen printouts, the additional source code was added so that each process writes the local times of its own invocations to the critical functions of the parallel system to a local file on disk. Figure 2 shows the layers of the I/O system where code instrumentation was implemented. Left boxes in blue, green and orange represent the layers on compute nodes. The bigger orange box depicts the layers on the storage nodes. Small orange boxes represent the I/O clients, which interact with the metadata and data servers (storage nodes). We have measured the times for the different functions called by ROMIO and PVFS2.

B. Execution Environment

One of the problems found in production systems is that the file system cannot be modified and instrumented [13]. Therefore, to create the HPC cluster where the entire I/O software stack with the embedded code instrumentation will be installed, Amazon's EC2 platform service was used. This type of platforms offer various types of instances based on the type of service purchased. In this case, the cluster was deployed using the free service and, even though these nodes offer very limited functionality as regards number of CPUs, memory, storage and network; they proved to be adequate to create the necessary environment for the tests we needed.

 ISBN: 978-1-61208-756-6

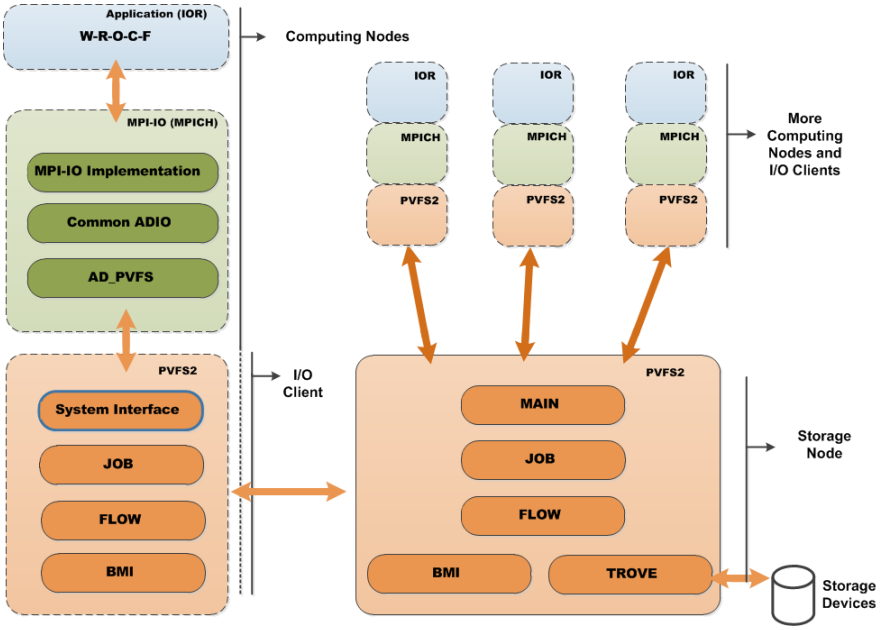


Figure 2: Code Instrumentation in the I/O Software Stack. Left boxes represent the layers on compute nodes and bigger orange box depicts the layers on the storage nodes.

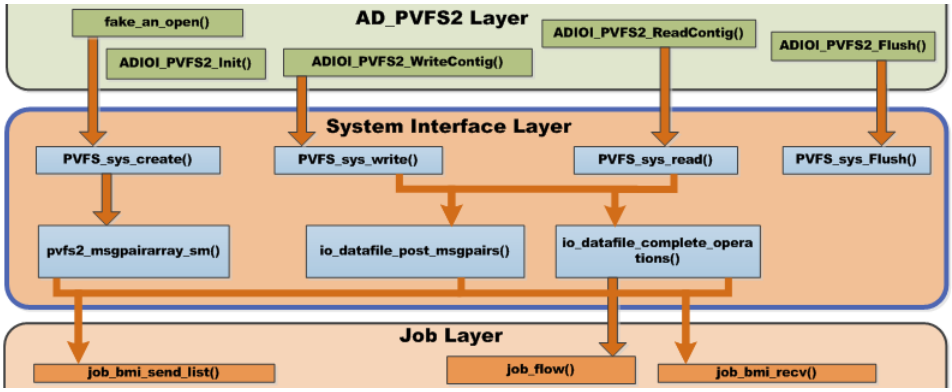


Figure 3: Selected functions of the System Interface layer.

Even though the execution environment affects the metrics obtained, the same configuration was run on a physical environment to compare it with that of Amazon. For instance, for a scenario with 8 computation nodes and I/O with one storage node, the times measured through code instrumentation in both environments are not the same, but they follow the same trend. The differences observed are mainly due to the different hardware performance in both execution environments.

Through the scenarios used, the critical functions involved in each layer of the I/O software stack were selected based on their role and execution time. As way of example, Figure 3 shows the functions selected in the System Interface layer of the I/O clients.

IV. MODELLING THE I/O SYSTEM

After analyzing each of the layers, a model of the I/O system was developed by implementing state machines and variables that describe each of those states. To that end, state machines were implemented for each of the layers in the system, differentiating their operation both on client and server side. The ultimate goal is using these state machines to design the behavior of each of the agents and its interactions with other agents and/or its environment.

The model developed is aimed to reproduce the interaction among the different components and analyzing how the information goes through the different modules or layers, with the possibility of measuring time to approach the real model of the I/O system. Therefore, each layer is modeled based on the execution flow of the functions that are called while processing certain requests, such as opening, closing, reading and writing operations. With the description of each function, the different states of the layers while carrying out those requests were implemented.

Due to the complexity to describe fully the modeling of the I/O software stack, we have selected the System Interface layer to explain in detail the calibration, verification and validation phases. Similar steps were done for the other layers.

The System Interface layer is a client-side interface that allows manipulating the objects in the file system. It launches a number of functions and state machines that process the operation in small steps.

 ISBN: 978-1-61208-756-6

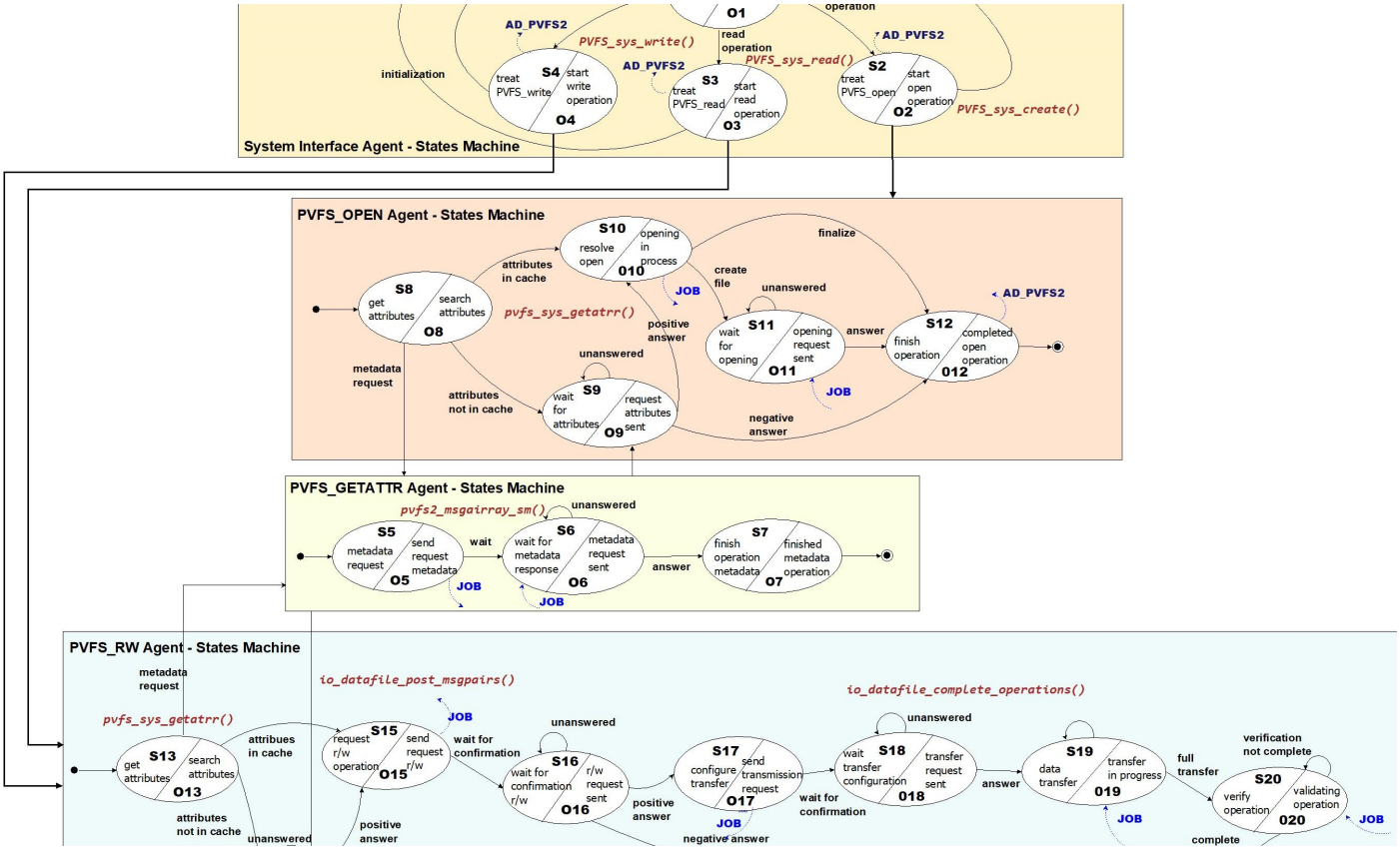


Figure 4: State machines for agents in the system interface layer.

In the context of PVFS, state machines execute a specific function in each of their states. The value returned by this function determines the state that should be adopted. Complex requests can be modeled; they are represented as a sequence of several states. Also, state machines can be nested to model and simplify common subprocess handling. These machines are used both in clients and servers.

There are several caches on the client side that are part of the System Interface layer and try to minimize the number of requests that the server has to process. The attributes cache (acache) manages metadata, while the name cache (ncache) stores the filename of file system objects and their respective handling number. To prevent caches from storing invalid information, data are set as invalid after a certain time has passed or when the server notifies the client that the object does not exist.

A. Functional Model

As shown in Figure 3, the functions in this layer are: `PVFS_sys_create()` to manage the creation of new files in the system, `PVFS_sys_write()` to perform writing operations, `PVFS_sys_read()` to perform reading operations, and `PVFS_sys_flush()` to dump file data to server storage. Each of these functions has internal variables and state machines that are run to carry out the relevant operations.

To simplify the model, we considered the following in relation to parallelism when handling several instances: **a)** I/O interfaces: layers `MPI-IO`, `ADIO` and `AD_PVFS` work in a sequential and blocking manner, since they run functions that require synchronization; this means that no instruction is served until the instruction being processed is completed. The calls run on their state machines are blocking; **b)** PVFS2 parallel file system: The System Interface, Job, Flow, BMI, Main Loop and Trove layers serve other requests and store their instructions in a buffer. Therefore, it allows handling different data flows.

The behavior of each of the agents is described by the state machine, the state transition table and the corresponding state variables. Figure 4 shows part of the state machines developed to model the operation of the System Interface layer, considering the functions and state machines corresponding to each of the three initial operations. As it can be seen, it consists of four agents called System Interface, which is responsible for decoding the instructions that enter the layer; `PVFS_OPEN`, which manages file opening operations; `PVFS_GETATTR`, which carries out searches in the metadata; and `PVFS_RW`, which manages file reading and writing operations.

The agent that manages file opening operations can only have one of five different states (S8 to S12). It will remain in S8 and configure agent `PVFS_GETATTR` if it requests metadata. If the attributes are not found in cache, it will transition to state S9 to wait for them; otherwise, it will transition to state S10. If in state S9, it will wait for a response from agent `PVFS_GETATTR` or it will complete the opening operation by communicating with the server, transitioning to state S10. If the operation cannot be completed, it will transition to state S12 to end.

While in state S10, it will start file creation through a request sent to the JOB layer, transitioning to state S11. Otherwise, it finishes the operation and transitions to state S12.

Figure 5: Simulator's user interface in NetLogo

While in state S11, it waits for a response to its file opening request and, if it receives one, it transitions to state S12. Once in state S12, it finishes the operation and sends a response to agent `AD_PVFS`.

On the other hand, agent `PVFS_RW` manages the write or read requests on client side. In Figure 4, there can be seen in red the functions selected that were used as the basis for the development of each state machine. For example, one of the functions belonging to `pvfs2_msgpairarray_sm()` [14], on which the `PVFS_RW` agent is based, is `io_datafile_post_msgpairs()` that is responsible for managing the data transmissions involved in the creation of files in agent System Interface. These communications occur, in the case of both a reading or writing, between client and server through the Job and BMI layers.

B. Initial model calibration

To obtain initial values for the functional model, we have monitored the selected functions for the IOR benchmark in a HPC cluster deployed in AWS EC2. The I/O system was configured over on PVFS2 parallel file system and the MPICH distribution. The cluster was composed by five nodes, where each one had three roles: compute node (computing and PVFS2 clients), metadata server and data server (datafiles). We have selected a simple pattern where file size and transfer size were updated. IOR was configured as follows:

- 1 GiB === `mpirun -np 5 ./ior -a MPIIO -b 205m -t 205m -F`
- 2 GiB === `mpirun -np 5 ./ior -a MPIIO -b 410m -t 410m -F`

For this setting, each process writes/reads to/from its own file in transfer sizes defined by the `-t` parameter. Due to the block size (`-b`) is equal to the transfer size (`-t`), only one operation is done by each process. The interface selected was MPI-IO for the *one file per process* (`-F`) strategy and independent I/O. The mapping corresponds to one MPI process per compute node.

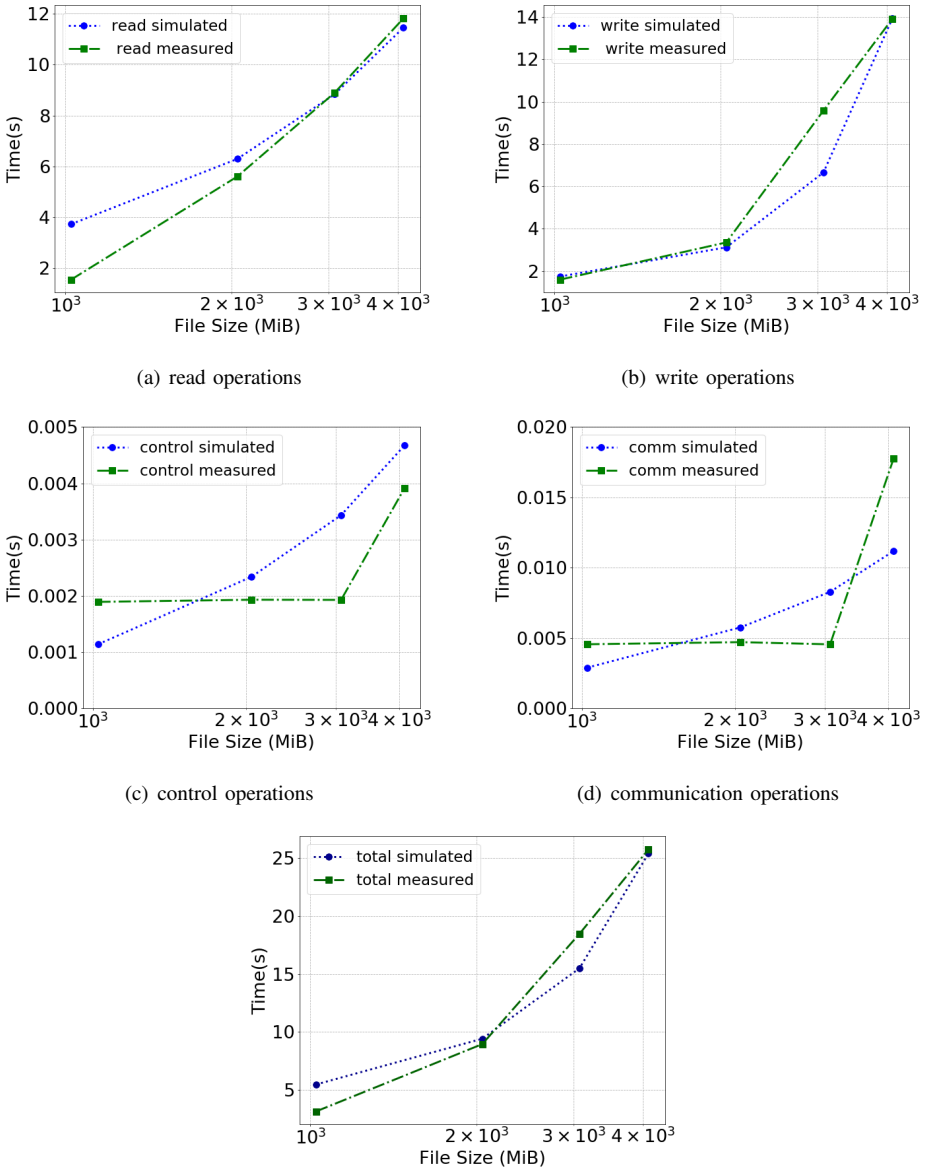
This measurement allows us to classify the monitored metrics in three groups: 1) *data access time* related with the data accesses operations such as write, read, and so on, 2) *control time* that includes verification and configuration of the data structures and 3) *communication time* related with the interaction between the clients and the metadata and data servers.

We have applied linear and exponential regressions for the time monitored in different functions of PVFS2. For this first analysis, we have selected as dependent variable the execution time and as independent variable the file size, request size is fixed for all the tests. In the case of the system interface layer, we have selected the following equations to represent the time of the functions:

- `PVFS_sys_create()` $= 0.0217 \times x$
- `PVFS_sys_write()` $= 0.8 \times e^{(0.7105 \times x)}$
- `PVFS_sys_read()` $= 2.5490 \times x + 1.2$
- `io_datafile_post_msgpairs()` $= 0.0012 \times x$
- `io_datafile_complete_operations()` $= 0.0028 \times x$

Where the x variable represents the file size to write or read. The statistical dispersion also depends on the file size and therefore it is calculated by using the same method.

 ISBN: 978-1-61208-756-6



(a) read operations

(b) write operations

(c) control operations

(d) communication operations

(e) total time

Figure 6: Simulated and Measured time in the system interface layer of the PVFS2

C. Initial Implementation

For the first proof of concept, an initial simulation model was developed using ABMS' framework called NetLogo. This framework includes a simplified programming language and a graphical interface that allows the user build, observe and use agent-based modeling without understanding complex standard programming language details. This tool is specifically indicated for the simulation of complex systems; it allows giving instructions to many independent agents that are concurrently executed, which is useful to study the connection between individual and collective behavior through agent actions and interactions.

The scenario adopted for the initial experiments was simulating the exchange of information among computation nodes and storage nodes, considering in each of them the layers discussed in previous sections. The MPI operations that can be served by the application layer are only I/O operations, and this initial implementation only includes open, read, write and close operations. One of the parameters allows toggling between executing only one type of operation or all of them. There is an option for selecting a maximum number of operations, which are distributed among the computation nodes selected.

The number of computation nodes and storage nodes can be configured. Node actions and interactions were fully implemented for the operations mentioned above. There are other parameters that allow selecting the existence of the data in the system before running the simulation, configuring the corresponding layers and preparing the I/O server for this scenario. Figure 5 shows the simulator's user interface. The configuration bars that the user has available to set the variables and parameters of the I/O software stack and the scenario to simulate are on the left. Also, the I/O configuration can be

made through command line. The center shows the distribution of the I/O system.

D. Verification and Validation

To validate the proposed model, we have configured a cluster in AWS EC2 similar to deployed in the calibration phase (see Section IV-B). The I/O system was deployed by using the PVFS2 parallel file system in a HPC cluster composed by five nodes, where each one was compute node (computing and PVFS2 clients), metadata server and data server (datafiles). IOR was executed for the following configurations:

- 1 GiB === `mpirun -np 5 ./ior -a MPIIO -b 205m -t 205m -F`
- 2 GiB === `mpirun -np 5 ./ior -a MPIIO -b 410m -t 410m -F`
- 3 GiB === `mpirun -np 5 ./ior -a MPIIO -b 615m -t 615m -F`
- 4 GiB === `mpirun -np 5 ./ior -a MPIIO -b 820m -t 820m -F`

Figure 6 presents the simulated and measured times for the IOR benchmark in the system interface layer of the PVFS2. As can be seen, the I/O behavior in this layer is dominated by the access data operations that corresponds to the read and write operations. The simulated total time of System Interface layer shows similar behavior to measured time but we can see that the control and communication time present a higher deviation for more than 3 GiB. However, due to the data access operations represent the highest I/O time we can consider the initial simulation model appropriate to represent the I/O behavior.

About the strange behavior of the communication and control functions, it can be observed in Figure 6(c) and 6(d) that these are constant for file size up to 3 GiB but these grows up for 4 GiB. We have also modeled the communication and control times by using the constant and exponential functions but these do not fix with the right behavior for the cases evaluated. Considering the evaluated pattern, these functions did not impact significantly on the I/O behavior but we must evaluate other file sizes and I/O patterns to can reduce the deviation and guarantee that these functions will have not impact on the I/O behavior independently of the data access pattern.

V. CONCLUSIONS

This paper presented a conceptual model of HPC I/O software stack by using agents based on state machines that act and communicate within the defined environments.

The instrumentation method used to obtain the different parameters for the simulator has been described. Likewise, the functionality found through the instrumentation of the system code was useful for the generation of state machines. This methodology allowed us to study the working of the system without the difficulty of obtaining exhaustive descriptions of the system, required by other modeling paradigms. An initial implementation using ABMS with NetLogo was validated for the IOR benchmark configured for sequential pattern, a single shared file, MPI-IO interface and independent I/O.

As future work, we will continue analyzing other techniques to monitor the data transfer rate (bandwidth) and the input/output operations per second (IOPs) at different levels of the I/O software stack. Furthermore, we will evaluate collective operations and other I/O strategies. Additionally, we will extend the model for the Lustre parallel file system.

ACKNOWLEDGMENT

This research has been supported by the Agencia Estatal de Investigación (AEI), Spain and the Fondo Europeo de Desarrollo Regional (FEDER) UE, under contract TIN2017-84875-P and partially funded by the Fundacion Escuelas Universitarias Gimbernat (EUG).

We thank Román Bond, research engineering of Universidad Nacional Arturo Jauretche (Argentina), for his support in the implementation of the simulator.

REFERENCES

[1] A. Núñez, J. Fernández, J. D. Garcia, F. Garcia, and J. Carretero, "New techniques for simulating high performance mpi applications on large storage net," J. Supercomput., vol. 51, no. 1, Jan. 2010, pp. 40–57.

[2] J. Kunkel, "Using Simulation to Validate Performance of MPI(-IO) Implementations," in Supercomputing, ser. Lecture Notes in Computer Science, J. M. Kunkel, T. Ludwig, and H. W. Meuer, Eds., no. 7905. Berlin, Heidelberg: Springer, 06 2013, pp. 181–195.

[3] N. Liu et al., "Modeling a leadership-scale storage system." in PPAM (1), ser. Lecture Notes in Computer Science, R. Wyrzykowski, J. Dongarra, K. Karczewski, and J. Wasniewski, Eds., vol. 7203. Springer, 2011, pp. 10–19.

[4] B. Feng, N. Liu, S. He, and X.-H. Sun, "HPIS3: Towards a High-performance Simulator for Hybrid Parallel I/O and Storage Systems," in Proceedings of the 9th Parallel Data Storage Workshop, ser. PDSW '14. Piscataway, NJ, USA: IEEE Press, 2014, pp. 37–42.

[5] C. Carothers, D. Bauer, and S. Pearce, "ROSS: a high-performance, low memory, modular time warp system," in Fourteenth Workshop on Parallel and Distributed Simulation., 2000, pp. 53–60.

[6] V. J. Julián and V. J. Botti, "Estudio de metodos de desarrollo de sistemas multiagente," Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial, vol. 7, 2003, pp. 65–80. [Online]. Available: http://www.redalyc.org/articulo.oa?id=92501806. Retrieve: 10/2019

[7] A. Borshchev and A. Filippov, "From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools," The 22nd International Conference of the System Dynamics Society, Oxford, England, 07 2004.

[8] E. Kremers, "Modelling and Simulation of Electrical Energy Systems through a Complex Systems Approach using Agent-Based Models," Ph.D. dissertation, Universidad del País Vasco (UPV/EHU), 2012.

[9] M. Taboada, E. Cabrera, F. Epelde, and E. Luque, "Using an agent-based simulation for predicting the effects of patients derivation policies in emergency departments," in International Conference on Computational Science, Barcelona, Spain, 2013, pp. 641–650.

[10] Sharcnet. Parallel I/O introductory tutorial. [Online]. Available: https://www.sharcnet.ca/help/index.php/Parallel_IO_introductory_tutorial. Retrieve: 10/2019. (2017)

[11] R. Thakur, W. Gropp, and E. Lusk, "Data Sieving and Collective I/O in ROMIO," in Proceedings of the 7th Symposium on the Frontiers of Massively Parallel Computation, ser. FRONTIERS '99. Washington, DC, USA: IEEE Computer Society, 1999, pp. 182–. [Online]. Available: http://dl.acm.org/citation.cfm?id=796733. Retrieve: 10/2019

[12] T. M. William Loewe and C. Morrone. IOR Benchmark. [Online]. Available: https://github.com/chaos/ior/blob/master/doc/USER_GUIDE. Retrieve: 10/2019. (2013)

[13] P. Gomez-Sanchez et al., "Using AWS EC2 as Test-Bed infrastructure in the I/O system configuration for HPC applications," Journal of Computer Science & Technology, vol. Volumen 16, no. 02, pp. 65-75, 11/2016 2016. [Online]. Available: http://hdl.handle.net/10915/57264. Retrieve: 10/2019

[14] PVFS2 Team, "PVFS 2 File System Semantics Document," PVFS Development Team, Tech. Rep., 2015.

 ISBN: 978-1-61208-756-6

A Consideration of Added Value Which Influences Information Diffusion

Yuya Ota
Soka University
Tokyo, Japan
Email: e18m5222@soka-u.jp

Norihiko Shinomiya
Soka University
Tokyo, Japan
Email: shinomi@soka.ac.jp

***Abstract*—In the case of existing information diffusion models, probability of information diffusion only follows parameters of neighbor nodes sending and receiving information. But, the momentum of diffusion gains in an exponential fashion whenever the information passes through influential nodes in a real network. For example, when information diffuses through social networking service (SNS), the speed of diffusion increases whenever information passes nodes that is influential users. This paper quantifies the degree of influence as information vitality, and proposes the model using it. In addition, the paper considers relationship between diffusion speed and information vitality in simulation of the model.**

Keywords–Information diffusion; Social network; Independent cascade model.

I. INTRODUCTION

Recent advances in communication technology have made it possible to increase individual information transmission capacity, thus social network was formed on the Internet. It enabled people to diffuse information wildly and rapidly without existing mass media. On the other hand, the anonymity of the network accelerates irresponsible information transmission; thus, social problems that are like fake news or hate speech are increasing. For example, fake news was used to attack political opponent and appeal the electorate in the American presidential election in 2016. There is large risk to change the result by wrong information. To solve the problem, we need to make mathematical model to predict information diffuse in the society, and find the factor to influence the diffusion.

Ample studies have proposed the information diffusion model, however, the model does not consider the change of added information value in the diffusion process because the probability depends completely on initial parameter that network nodes and edges have. One of the value is diffusion trend that is determined by the contents of the information.The paper by Vosoughi et al. [1] have concluded that wrong information spread 20 times faster than right one. This means that the speed of information diffusion depends on the content. Another study analyzes the big contents data that is collected from SNS to find the diffusion trend [2]. These studies elucidate how the content influences the diffusion process; however, it cannot lead to predict the information diffusion.

The purpose of our study is making the information diffusion model to explain how the change of added information value influences the diffusion process. To test the hypothesis that the information diffusion depends on the value change in the process, we have examined whether the centrality of network for information diffusion has correlation with range of the diffusion. To this end, we have performed a series of different simulation with stochastic model.

In Section 2, information diffusion model using added information value is explained. In Section 3, method of simulations and results are illustrated. Finally, considering and future work were suggested in Section 4.

II. MODEL REPRESENTATION

This model considers the added information value, for example the number of re-tweet on Twitter, as the factor which influences the diffusion process. As shown in Table 1, these kinds of parameter can be considered as the added information value. The value is defined by information engine theory as the name of "information vitality (IV)" [3]. The first question which we have to answer in this paper is how the IV works in the information diffusion process.

Information engine theory has supposed that information has two parameters which are entropy and IV. Entropy is a parameter which is used in thermodynamics and statistical mechanics and considered as a physical quantity that expresses "ambiguity" in information theory. Information which has higher entropy is more likely to diffuse, so entropy and initial IV have a correlation.

In the IV model, we have used directed graph, and regarded the node receiving the information as active node, the ones not receiving node as inactive node, and the human relationship as edges. We have also considered the IV of the active nodes as diffusion probability $p_{u,v}$ $(0 \leq p_{u,v} \leq 1)$. When a node becomes active, the node tries to make neighbor nodes active only once in the next step. In addition, every node has the parameter named "logical work" that change IV. When information passes through the nodes, each node gives logical work q $(q \in \mathbb{R})$ to the information, and change the IV. Therefore, when a node becomes active in the probability of $p_{u,v}$, the node tries to diffuse in the probability of $p_{u,v} + q$ in next step. Although the probability of information diffusion does not change in previous independent cascade (IC) model as figure 1 shows, in the case of Figure 2, initial IV is 0.2, and when it makes next node active, it becomes 0.21. This simulation keeps going until the state does not change. This model can explain spread of news or advertisement on SNS as information initiative model.

III. SIMULATION

We have performed diffusion simulation to assess the performance of proposed model using large network of news communication web site. For comparison, we have also performed simulation using IC model.

The network is a directed graph, and has 30398 nodes and 87627 arcs. We chose one node which has enough diffusing capacity as the starting node. In the case of IC model, we set 20% diffusion probability for every arc, whereas in the case

 ISBN: 978-1-61208-756-6

TABLE I. ADDITIONAL INFORMATION PROPERTY

	Added Value
Facebook	Number of Likes
Twitter	Number of Re-tweets
YouTube	Number of Views
Text	Language
General information	Security parameter

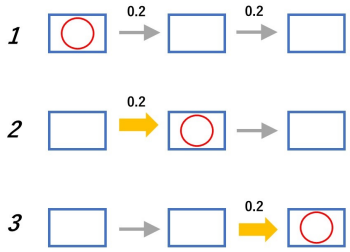


Figure 1. Concept of IC model

of IV model, the diffusion probability is increasing by logical work of every node which the information has passed.

Figure 3 shows that when the diffusion ends before it spreads widely, there is little difference between two models. The only difference is that threshold to cause explosive diffusion became high because of increasing IV. Next, Figure 4 shows the data of explosive diffusion. The simulation data demonstrates that IV changes the explosive diffusion wide and high frequency.

The second simulation has been designed to assess degree of the diffusion suppression per three types of centrality including degree centrality, page-rank, and personalized page-rank. Only personalized page-rank rates the closeness of the nodes to the starting node. We sorted all nodes according to each centrality, then decreased the probability of the nodes in order of the centrality. The simulation investigated how distance from the starting node influences to the centrality of the node in aspect of information diffusion. Results of the simulation, as illustrated in Figure 5, when we use personalized page-rank, information diffusion is well suppressed, and variation

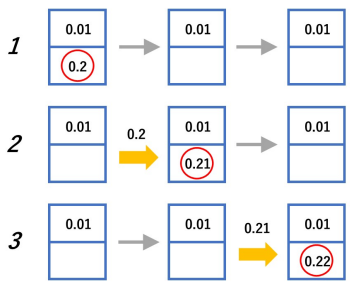


Figure 2. Concept of Information vitality model

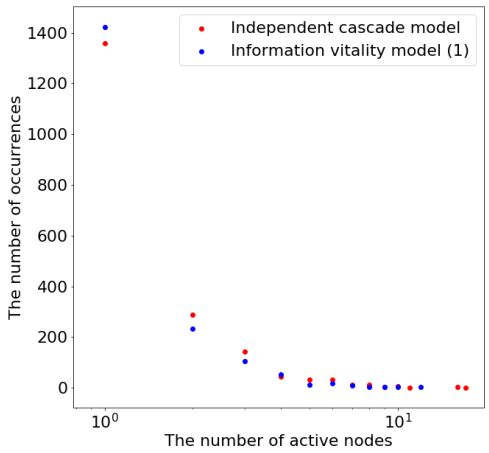


Figure 3. Scatter plot for small-scale diffusion (less than 1000 nodes)

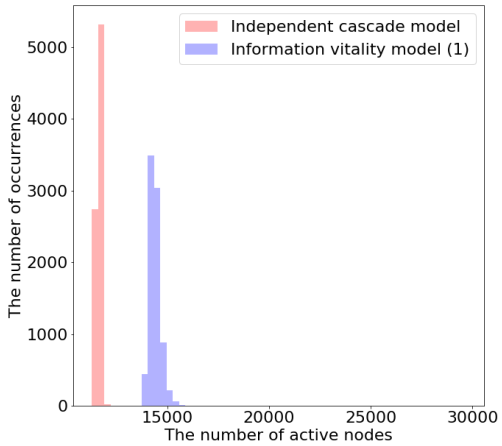


Figure 4. Histogram for Large-scale diffusion (more than 1000 nodes)

rate is the least in these types of centrality. It is reasonable to support that early stage of information diffusion has big chance to change conclusive wideness of diffusion, and it is more important to weaken the nodes which are close to the starting node than the nodes which have high degree.

IV. CONCLUSION

In order to make the model that explains information diffusion phenomena especially in social networks on the Internet, we quantified the added information value as IV. Then, we made the information diffusion model that the diffusion probability changes according to the diffusion process. As a result, it was found that in the case of small-scale diffusion with less than 1000 active nodes, the influence of added information value remained on the change of threshold leading to large-scale diffusion.

On the other hand, in the case of large-scale diffusion with 1000 or more active nodes, the range of diffusion increased more than the case that the added information value was not

 ISBN: 978-1-61208-756-6

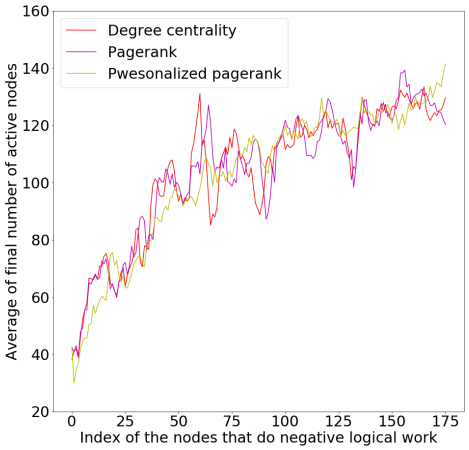


Figure 5. Diffusion suppression per each centrality

taken into account. Information such as the number of views on YouTube can be considered to have the effect to expand the range of spreads explosively after the video spread more than a certain range. In order to accurately model the actual information diffusion which is determined by human, more complicated algorithm will be necessary. We will examine and work on the construction of the more accurate information diffusion model.

REFERENCES

[1] S. Vosoughi, D. Roy, and S. Aral, The spread of true and false news online, Science: Vol. 359, Issue 6380, pp. 1146-1151, March 2018.

[2] K. Ohara, R. Odagiri, and S. Shirakawa, On a Topic-based Information Diffusion Model, SIG-KBS/SIG-KBS,B5(01) pp. 13-17, 2015.

[3] H. Watanabe, "A Basic Theory of Information Network", IEICE TRANS FUNDAMENTALS, VOL.E76-A, NO.3, pp.265-276, March 1993.

 ISBN: 978-1-61208-756-6

Objective Evaluation of a Novel Filter-Based Motion Cueing Algorithm in Comparison to Optimization-Based Control in Interactive Driving Simulation

Patrick Biemelt, Sven Mertin, Nico Rüddenklau, Sandra Gausemeier and Ansgar Trächtler
Chair of Control Engineering and Mechatronics, Heinz Nixdorf Institute, University of Paderborn, Germany
Email: {patrick.biemelt, sven.mertin, nico.rueddenklau, sandra.gausemeier, ansgar.traechtler}@hni.uni-paderborn.de

***Abstract*—Dynamic driving simulators have become a key technology to support the development and optimization process of modern vehicle systems both in academic research and in the automotive industry. However, the validity of the results obtained in simulator tests depends significantly on the adequate reproduction of the simulated vehicle movements and the associated immersion of the driver. Therefore, specific motion platform control strategies, so-called *Motion Cueing Algorithms* (MCA), are used to replicate the acting accelerations and angular velocities within the physical limitations of the driving simulator best possible. In this paper, we present a novel filter-based control approach for this task, using a hybrid kinematics motion system as an application example. Based on introduced quality criteria, an objective comparison of the proposed control strategy and a real-time capable *Model Predictive Control* (MPC) algorithm is performed using various standard driving scenarios. These include longitudinal as well as lateral dynamic maneuvers in order to estimate the overall improvements of both Motion Cueing Algorithms for interactive driving simulation.**

Keywords–Driving Simulation; Human-in-the-Loop; Motion Cueing; Dynamic Motion Platform Control; Objective Quality Criteria.

I. INTRODUCTION

Driven by topics, such as e-mobility and autonomous driving, in recent years there has been a continuous trend towards interconnectivity and multifunctionality of vehicle components, as well as Advanced Driver Assistance Systems (ADAS). As a consequence, automobile manufacturers and developers have to deal with increased product complexities and simultaneously decreased development periods to ensure their competitiveness in the automotive industry. To overcome those new technological challenges, the use of interactive driving simulators represents an indispensable tool to transfer the conventional development process, based on physical prototypes and on-road tests, to model-based test procedures. Such virtual prototyping methods using driving simulators provide the benefit of time and cost savings, as well as safe and reproducible test environments with a high level of flexibility at the same time. For example, varying weather and lighting conditions can be directly adapted to the test requirements in the simulated environment, which supports i.a. the development and optimization of modern headlamp systems significantly [1]. Furthermore, interactive driving simulation enables to access human-centered aspects, such as demonstration and marketing, driver training and behavior studies [2][3].

Disregarding from the particular analysis purpose, the validity of the results obtained in a virtual test drive is closely linked to the degree of immersion. Interactive driving simulation can therefore be characterized as a *Human- and Hardware-in-the-Loop* (HHiL) application whose transferability to real driving situations can only be guaranteed if a realistic driving impression is created. Hence, it is necessary to provide the human perception system with all required motion information, so-called *Motion Cues*. In addition to the acoustic and visual stimuli, also the vestibular Motion Cues, more precisely the acting translational accelerations and angular velocities of the simulated vehicle, must be generated using the motion system of the dynamic driving simulator. For this reason, specific Motion Cueing Algorithms are applied in order to create a driving experience that is as realistic as possible within the physical limitations of the motion system.

The most common approach for this task is the *Classical Washout Algorithm*, which was first described by Schmidt and Conrad as a motion platform control algorithm for piloted flight simulators [4]. As illustrated in Figure 1, this MCA basically consists of a sequence of frequency divisions in order to generate suitable position and orientation reference signals for the simulator motion system. The high-frequency components of the scaled translational accelerations and angular velocities of the vehicle dynamics model are therefore separated using appropriate high-pass filters. Afterwards, these extracted components are directly integrated to a corresponding position and orientation of the driving simulator. Since the basic idea of this algorithm is to return the motion system to its neutral position after it has performed the high-frequency movements, a further high-pass filtering of the integrated signals is conducted. This is known as the *washout effect*, according to which the algorithm is named. Due to the typically small workspace, an analog integration of the low-frequency accelerations and angular velocities would lead the motion system very fast to its physical limits and thus cannot be performed. Hence, sustained accelerations are simulated via the *tilt coordination* technique, which makes use of the gravitational force to replicate these accelerations by an equivalent rotation of the driving simulator. The corresponding rotation rate is usually limited to the perception threshold of the human vestibular system in this process, so that the rotational motion will not be realized by the driver inside the simulator.

This simple control strategy has been extensively studied and improved since its first publication, typically using

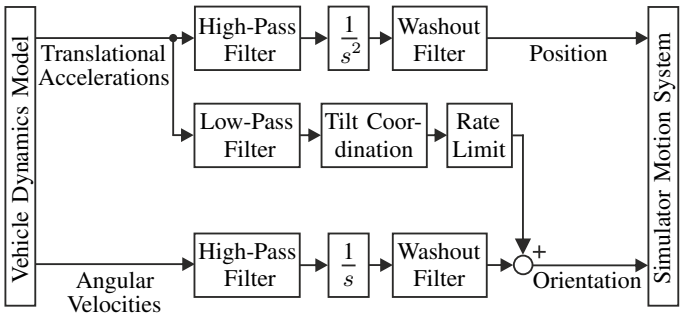


Figure 1. Scheme of the Classical Washout Algorithm [4].

hexapod-based motion systems [5][6]. As a result of this research, the filter-based MCA evolved into the standard approach in interactive driving simulation that offers major benefits in terms of transparency and traceability. Each parameter in the Classical Washout Algorithm has a clear physical meaning and a unique association to a single degree of freedom, which simplifies the tuning significantly. However, this basic idea of treating the translational accelerations and angular velocities independently results in the fact that this approach cannot be applied to every type of motion system. Otherwise, conflicting vestibular stimuli are generated under certain circumstances, e.g., if there exist interdependencies between translational and rotational degrees of freedom of the motion system like it is introduced in the next section with the ATMOS driving simulator.

In the present work, we propose a novel filter-based Motion Cueing Algorithm that enables a dynamic position washout to any point within the simulator workspace without considerably affecting the high-frequency motion rendering. This key feature is motivated by the considered motion system, but can also be applied to other systems, which offers general advantages for interactive driving simulation. The resulting control quality is evaluated by means of defined objective quality criteria, which take into account both measured and perceived quantities, including models of the human perceptual system. Based on this valuation metric, the developed filter-based MCA is compared online to a Model Predictive Control approach using established driving scenarios from the automotive industry, as well as everyday driving maneuvers.

The rest of this paper is structured as follows: Section II briefly describes the considered motion system and its specific features that have to be taken into account to ensure a realistic driving impression. In Section III, the developed filter-based control strategy is presented in detail. Subsequently, the objective valuation metric and the examined driving scenarios are introduced in Section IV. Sections V and VI finally discuss the obtained results and give concluding remarks.

II. ATMOS Dynamic Driving Simulator

Figure 2 shows the Atlas Motion System (ATMOS) driving simulator that is operated at the Heinz Nixdorf Institute in Paderborn as a reconfigurable development platform for primarily lighting-based ADAS. As illustrated, this simulator is equipped with a real vehicle chassis including all its control actuators, a seamless circular projection with 270 degree viewing angle, a 5.1 multichannel audio system, as well as a unique five-degree-of-freedom motion system to guarantee full immersion of the driver in the virtual environment. Moreover, the acting accelerations and angular velocities are recorded using an Inertial Measurement Unit (IMU) that is installed close to the driver's head position in order to rate the quality of the applied Motion Cueing strategy.

Different from conventional hexapods [7], the motion system of the ATMOS driving simulator is designed as a hybrid kinematics system, which is composed of two mechanically coupled components that can be actuated independently. To illustrate the functionality, Figure 3 shows an exploded view based on the multibody model of the system. The shaker system below the vehicle chassis is equipped with three crankshaft drives to perform vertical translational movements, as well as to rotate the driver around the roll and pitch axis. Thus, the shaker replicates the simulated vehicle movements

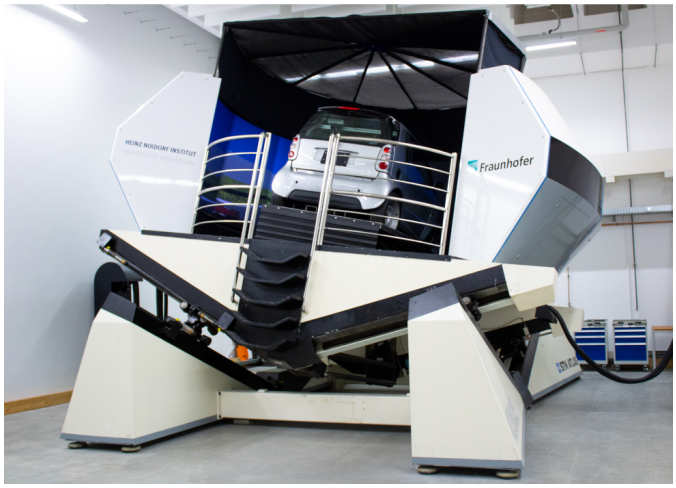


Figure 2. ATMOS Dynamic Driving Simulator.

relative to the road surface with exception of yaw movements and can further be used to increase the effect of the tilt coordination by expanding the rotational workspace of the motion system. In addition to the shaker, the motion platform performs movements along curved axes in lateral and longitudinal direction via actuated cross-undercarriages that are driven on V-shaped tracks. Because of these tilted tracks, each translational movement of the motion platform leads simultaneously to an additional rotation around the corresponding axes and a vertical displacement of the platform center point. As a direct consequence of these coupled kinematics, performing pure translational movements of the motion platform is not possible. This has to be considered in the design of the control algorithm in order to avoid conflicting sensory information, so-

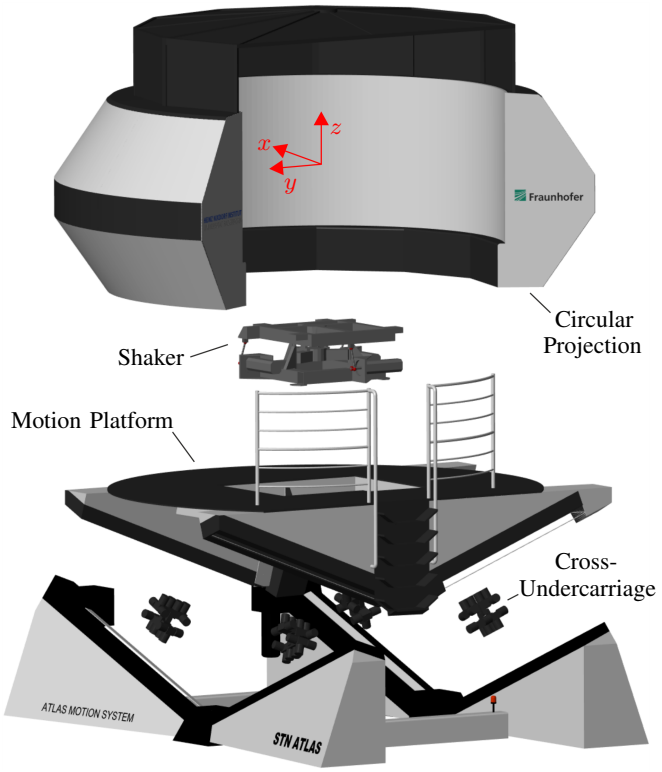


Figure 3. Exploded View of the Simulator Multibody Model.

called *False Cues*, which typically lead to the undesired effect of *Simulator Sickness* for the driver [8].

Due to the mentioned features of the ATMOS driving simulator, an extension of the conventional filter-based MCA is required since the implementation of the Classical Washout Algorithm according to Figure 1 does not result in the desired quality of the interactive driving simulation.

III. Developed Washout Algorithm

As described in Section I, the general idea of the Classical Washout Algorithm is based on an independent consideration of the systems degrees of freedom, which is due to the fact that the MCA was developed for application on a conventional hexapod. Because of this, the algorithm cannot be transferred to the ATMOS driving simulator introduced in the previous section, as there is a connection between translation and rotation because of the underlying kinematics of the motion system. For this reason, we subsequently present an extension of the classical approach that includes the relevant kinematic effects and enables a sufficient control quality.

A. Dynamic Position Washout

In case of the regarded driving simulator, each longitudinal and lateral movement of the motion platform generates a forced tilting around the corresponding roll and pitch axis. These rotations should ideally be used to emulate sustained accelerations using the tilt coordination technique. Otherwise, the tilt coordination has to be performed only by the shaker, which limits the maximum possible inclination to the small shaker workspace. In contrast to the classical algorithm, a dynamic position washout is therefore required that enables the motion platform to drift into a defined end position within its workspace after it has performed the high-frequency movements. By determining this end position according to the associated inclination, low-frequency accelerations can also be simulated via the motion platform. For this purpose, the high-pass (hp) and washout (wo) filters of the high-frequency longitudinal and lateral acceleration paths are supplemented by further first order low-pass filters with variable gains K, as shown in Figure 4 using the example of longitudinal acceleration a_x. The non-intuitive idea of this extension can be clarified by the application of the final value theorem of the Laplace transform. Therefore, let a_x be a sustained acceleration input from the vehicle dynamics simulation, which can be assumed to be approximately constant, since the magnitude does not significantly change. For the integrated simulator position x applies then with increasing time t:

$$\begin{aligned}\lim_{t\to\infty} x(t) &= \lim_{s\to 0} s \cdot X(s) \\ &= \lim_{s\to 0} s \cdot \frac{T_{hp}s + K}{T_{hp}s + 1} \cdot \frac{1}{s^2} \cdot \frac{T_{wo}^2 s^2}{T_{wo}^2 s^2 + 2DT_{wo}s + 1} \cdot \frac{a_x}{s} \\ &= K \cdot T_{wo}^2 \cdot a_x\end{aligned} \tag{1}$$

This yields a resulting simulator position x that depends on the gain K, the time constant T_{wo} of the washout filter as well as the amplitude of the acting acceleration a_x. If this position is now required to have a defined value x_{tc}, the necessary gain K can be determined corresponding to (1) as

$$K = \frac{x_{tc}}{T_{wo}^2 \cdot a_x} \,. \tag{2}$$

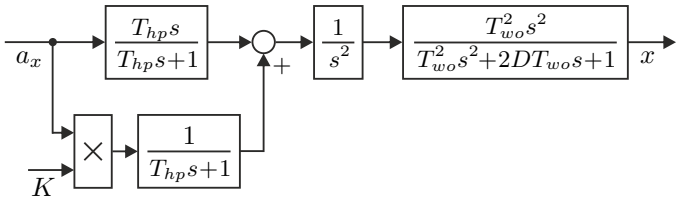


Figure 4. Extended Longitudinal High-Frequency Acceleration Path.

Analogously, the initial value theorem of the Laplace transform can be used to show that the extension by the variable gain low-pass filter, as shown in Figure 4, does not negatively affect the reproduction of high-frequency acceleration components. Like in the Classical Washout Algorithm, the dynamics of the drift into the end position x_{tc} can be influenced by the parameters of the washout filter, which is an additional design freedom in the parameterization of the developed algorithm.

The described extension is also implemented for the lateral high-frequency acceleration path, so that a washout in the defined position y_{tc} according to (2) is realized and thus sustained lateral stimuli are produced by a corresponding roll rotation of the motion platform.

B. Tilt Coordination Distribution

Due to the hybrid kinematics motion system, as well as the presented dynamic position washout, the tilt coordination technique can be performed either using the motion platform (mp), the shaker (sh) or a combination of both systems. The latter significantly increases the workspace and thus the maximum low-frequency acceleration amplitudes that can be generated. Consequently, a distribution strategy has to be specified, which enables a suitable coordination of both components. For this reason, an adaptation of the low-frequency longitudinal and lateral acceleration paths is conducted according to Figure 5. As shown with the example of the longitudinal acceleration, a first order low-pass (lp) filter extracts the sustained acceleration components from the reference signal a_x, which are subsequently converted to the corresponding tilt coordination pitch angle θ_{tc}. In doing so, the associated rotation rate is limited to the well-established value of $0.1\,rad/s$, in order that the tilt coordination is not detected by the human driver [9]. In contrast to conventional hexapods, this inclination is divided among the subsystems of the motion system by introducing a distribution coefficient $\alpha \in \mathbb{R}$ with $0 \leq \alpha \leq 1$. This results in the inclinations for the shaker θ_{sh} and for the motion platform θ_{mp} that are necessary to replicate the low-frequency accelerations by the gravitational force. Based on the kinematic relations of the motion platform, an equivalent platform position x_{tc}, which corresponds to the required inclination, is

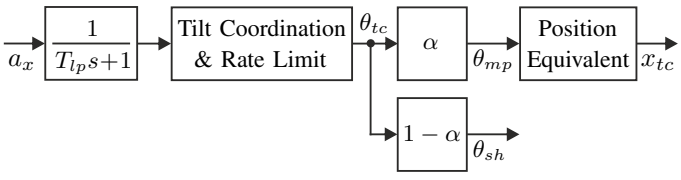


Figure 5. Extended Longitudinal Low-Frequency Acceleration Path.

 ISBN: 978-1-61208-756-6

subsequently determined. This position equivalent then serves as input for calculating the variable gain K according to (2) so that the coupling between translational and rotational degrees of freedom is taken into account. Equally, this process is implemented for the lateral low-frequency acceleration path.

C. Resulting Algorithm Structure and Parameterization

The combination of dynamic position washout and tilt coordination distribution leads to the overall structure of the developed washout algorithm illustrated in Figure 6. Based on the idea of the Classical Washout Algorithm, this filter-based control strategy enables the generation of suitable control signals in the form of position and orientation commands for the motion system of the ATMOS driving simulator. The necessary estimation of the corresponding filter parameters and distribution coefficients was performed by numerical optimization using a defined driving maneuver. Here, the rural road drive, which will be introduced in the next section, was chosen since it represents a good compromise between moderate driving situations and extreme maneuvers at the limits of driving dynamics. Table I provides an overview of the resulting parameters.

Although the developed algorithm is motivated by the specific features of the motion system, in particular the concept of a dynamic position washout offers great potential for application in alternative control approaches. Thus, an integration in the context of lane-based algorithms is feasible, which use vehicle position information on the road to laterally preposition the motion system and therefore use the available workspace more effectively [6][10].

TABLE I. APPLIED ALGORITHM PARAMETERS.

Scaling	*1st Order HP Filter*	*1st Order LP Filter*	*2nd Order WO Filter*	*Distribution Coefficient*
$k_x = 0.4$	$T_{hp} = 0.95$	$T_{lp} = 0.95$	$T_{wo} = 0.49$, $D = 0.7$	$\alpha_x = 0.65$
$k_y = 0.4$	$T_{hp} = 0.6$	$T_{lp} = 0.6$	$T_{wo} = 0.44$, $D = 1.0$	$\alpha_y = 0.6$
$k_z = 1.0$	$T_{hp} = 0.4$	–	$T_{wo} = 0.45$, $D = 1.0$	–
Scaling	*1st Order HP Filter*	*1st Order LP Filter*	*1st Order WO Filter*	*Distribution Coefficient*
$k_\varphi = 1.0$	$T_{hp} = 1.2$	–	$T_{wo} = 0.8$	–
$k_\theta = 1.0$	$T_{hp} = 0.3$	–	$T_{wo} = 0.2$	–

IV. COMPARISON OF THE CONTROL STRATEGIES

The scientific objective in this contribution deals with the comparison of the filter-based MCA presented in Section III with an optimization-based approach using the concept of Model Predictive Control. This real-time capable predictive controller results from previous research and is described in detail in [11]. For that reason, only the basic idea of the algorithm is briefly discussed in this section. In addition, the applied quality criteria as well as the examined driving scenarios are introduced afterwards.

A. Model Predictive Control Approach

According to the general concept of the MPC paradigm, an optimal control problem is numerically solved over a receding time horizon at each calculation cycle. Subsequently, only the first element of the computed control variable is applied to the process and the procedure is iterated. The application of this technique in terms of a Motion Cueing Algorithm leads to the structure shown in Figure 7. Based on the current system state $x \in \mathbb{R}^{15}$, which contains the angles, the angular velocities and the angular accelerations of all five actuators, the output vector $y = \begin{bmatrix} a^T & \omega^T \end{bmatrix}^T \in \mathbb{R}^5$ is estimated within the prediction horizon N, depending on the reference angles of each actuator $u \in \mathbb{R}^5$. The MPC determines suitable system inputs so that the predicted outputs best possible match the vector of the reference accelerations and angular velocities $r \in \mathbb{R}^{5N}$ from the vehicle dynamics simulation. In this approach, the accuracy of the control algorithm depends significantly on the availability of an adequate process model to predict the future system behavior. In case of the presented controller, the prediction model includes a linear model of the actuator dynamics, as well as the nonlinear kinematic relations of the motion system. This ensures that the relevant characteristics of the considered motion system described in Section II and its effects on the driver in the simulator are taken into account in the optimization process. At the same time, it is guaranteed that the determined actuating variables can be realized by the motion system, since its physical limits are included as constraints in the optimization problem. In that context, a first order approximation of the nonlinear system behavior at runtime presents a key feature of the developed predictive controller to meet the real-time requirements, which is the primary challenge in optimization-based MCA.

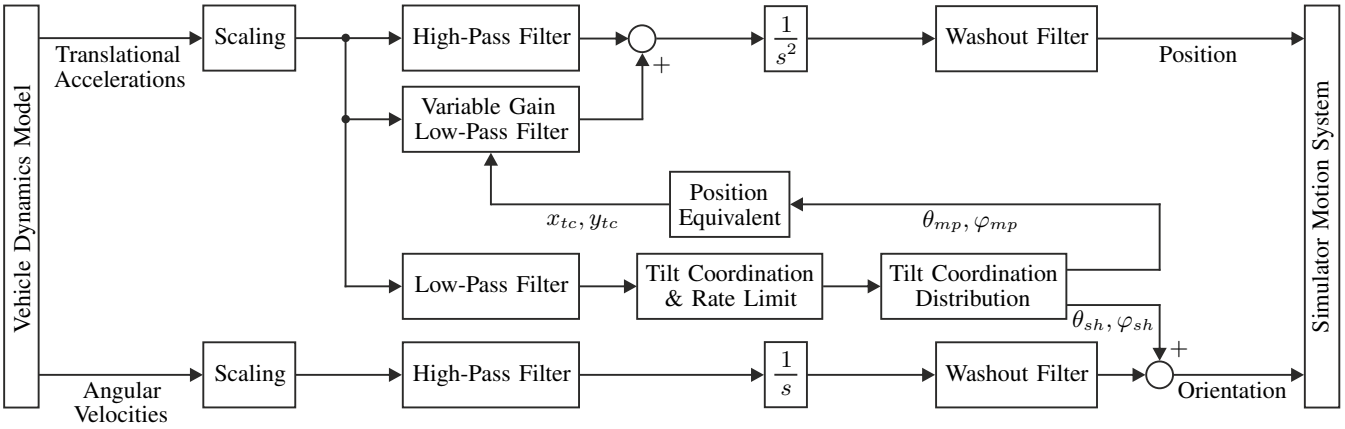


Figure 6. Overall Structure of the Developed Washout Algorithm.

 ISBN: 978-1-61208-756-6

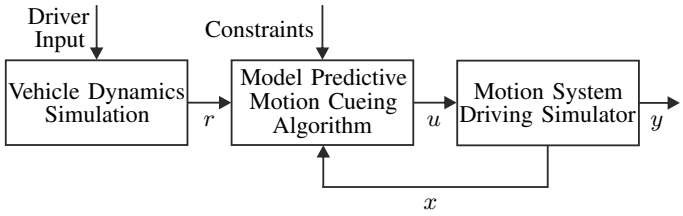


Figure 7. Scheme of the MPC-Based Control Algorithm.

B. Objective Quality Criteria

In order to compare both Motion Cueing strategies on the basis of an objective valuation metric, suitable quality criteria must be specified. Therefore, according to [12] and [13], we introduce performance indicators λ_1 and λ_2 that are defined as

$$\lambda_1 = \frac{1}{N}\sum_{j=0}^{N}\sqrt{\left(\frac{e_{a_x,j}}{a_{x,norm}}\right)^2+\left(\frac{e_{a_y,j}}{a_{y,norm}}\right)^2+\left(\frac{e_{a_z,j}}{a_{z,norm}}\right)^2} + \frac{1}{N}\sum_{j=0}^{N}\sqrt{\left(\frac{e_{\omega_x,j}}{\omega_{x,norm}}\right)^2+\left(\frac{e_{\omega_y,j}}{\omega_{y,norm}}\right)^2} \tag{3}$$

and

$$\lambda_2 = \frac{1}{N}\sum_{j=0}^{N}\sqrt{\left(\frac{e_{\hat{a}_x,j}}{a_{x,norm}}\right)^2+\left(\frac{e_{\hat{a}_y,j}}{a_{y,norm}}\right)^2+\left(\frac{e_{\hat{a}_z,j}}{a_{z,norm}}\right)^2} + \frac{1}{N}\sum_{j=0}^{N}\sqrt{\left(\frac{e_{\hat{\omega}_x,j}}{\omega_{x,norm}}\right)^2+\left(\frac{e_{\hat{\omega}_y,j}}{\omega_{y,norm}}\right)^2} \tag{4}$$

with

$$\begin{aligned} e_{a_i} &= a_{i,Ref} - a_i|_{i=x,y,z} \quad \text{and} \quad e_{\omega_i} = \omega_{i,Ref} - \omega_i|_{i=x,y} \\ e_{\hat{a}_i} &= \hat{a}_{i,Ref} - \hat{a}_i|_{i=x,y,z} \quad \text{and} \quad e_{\hat{\omega}_i} = \hat{\omega}_{i,Ref} - \hat{\omega}_i|_{i=x,y}. \end{aligned} \tag{5}$$

Here, (3) provides a measure of the physical deviations between the scaled reference accelerations $a_{i,Ref}$ and angular velocities $\omega_{i,Ref}$ from the vehicle dynamics simulation and the measured quantities in the driving simulator for the considered degrees of freedom. λ_1 therefore returns the averaged normalized control error over the number of measured values N. The normalization is necessary to obtain dimensionless quantities that allow a simultaneous consideration of accelerations and angular velocities in a common scale. According to [14], the human perception thresholds for movements are used as corresponding normalization factors $a_{i,norm}$ and $\omega_{i,norm}$. In addition, the indicator λ_2 according to (4) yields a measure for the perceived control quality, which can differ from the physical deviations due to the frequency-dependent dynamic behavior of the human vestibular organs, as well as perception thresholds. This causes, for example, that control errors in detectable frequency ranges are perceived more disturbing than deviations in undetectable ranges. To take these effects into account, well-established models of the vestibular system, more precisely the otoliths and semicircular canals, are included as they are shown in Figure 8. These usually contain mechanical analogous models of the respective organs, which lead to the illustrated transfer functions with the inputs a_i and ω_i [15][16], as they are widely used in driving simulation applications [9].

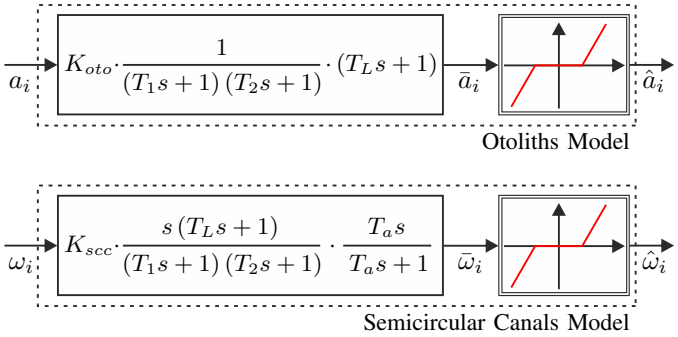


Figure 8. Applied Models of the Human Vestibular System.

By a series connection of the transfer functions with nonlinear dead zones, the threshold values $a_{i,thres}$ and $\omega_{i,thres}$ of the human perception are integrated with respect to the following relationship [5]:

$$\begin{aligned} \hat{a}_i &= \begin{cases} 0 & \text{if } |\bar{a}_i| \le a_{i,thres} \\ \bar{a}_i - sgn(\bar{a}_i)\cdot a_{i,thres} & \text{if } |\bar{a}_i| > a_{i,thres} \end{cases} \\ \hat{\omega}_i &= \begin{cases} 0 & \text{if } |\bar{\omega}_i| \le \omega_{i,thres} \\ \bar{\omega}_i - sgn(\bar{\omega}_i)\cdot \omega_{i,thres} & \text{if } |\bar{\omega}_i| > \omega_{i,thres} \end{cases} \end{aligned} \tag{6}$$

Consequently, the closer the performance indicators λ_1 and λ_2 are to the origin, the better is the reproduction of the simulated vehicle movements, whereby the value zero indicates a perfect motion rendering. However, especially with regard to λ_1, this is only a theoretical value that cannot be obtained by any driving simulator, since it would require an almost unlimited workspace.

C. Driving Scenarios

For the purpose of obtaining a representative comparison of the two control strategies, a selection of nine driving scenarios was defined. These contain standardized maneuvers, which are commonly used for development and optimization applications in the automotive industry, like:

- Acceleration from standstill
- Braking from driving straight forward (DIN ISO 70028)
- Lane change (DIN ISO 3888-1)
- Step steering (DIN ISO 7401)
- Braking from steady-state circular course drive (DIN ISO 7975)

As the listed maneuvers are mainly used to identify and analyze the driving dynamics of a vehicle, they do not represent usual driving situations. For this reason, also moderate scenarios are examined in the evaluation:

- Turning at a junction
- Drive on a rural road
- Drive through a roundabout
- Drive through a highway interchange

Vehicle dynamics simulations of all nine maneuvers were performed and the relevant accelerations and angular velocities were recorded. Subsequently, these data were used as identical reference signals for both MCA to ensure a consistent basis for evaluation described in the next section.

 ISBN: 978-1-61208-756-6

V. Results and Discussion

Subsequently, the results of the comparison of the two Motion Cueing strategies are presented and the impacts on the interactive driving simulation are discussed. For that purpose, both control algorithms were implemented on the ATMOS driving simulator. Measurement data of the translational accelerations and the angular velocities taken with the installed IMU at the driver's head position serve as inputs for the quality criteria presented in Section IV. For reasons of clarity, only measured data of the maneuver "turning at a junction" are analyzed in detail (see Figures 9, 10, 11). All further driving scenarios are summarized in Figure 12.

Figure 9 shows the resulting longitudinal acceleration and pitch velocity tracking. It becomes clear that both the proposed washout algorithm and the optimization-based approach yield an adequate reproduction of the longitudinal acceleration from the vehicle dynamics simulation. However, the measured accelerations show a larger time delay in comparison to the reference signal when using the washout algorithm due to the phase shift of the implemented filters. The corresponding pitch velocity contains in both cases low-frequency disturbances that can be explained by the tilt coordination, since the sustained acceleration can only be reproduced by an equivalent rotation of the motion system. Using the washout algorithm, these errors are significantly higher, so it can be expected that the resulting driving experience will be negatively affected. In contrast, the predictive MCA uses the available model knowledge to limit the overall rotation rate error to the value of $0.1\,rad/s$. Equivalent results can be derived from Figure 10, which illustrates the lateral acceleration and the corresponding roll velocity. Also in this case, the acceleration reference from the vehicle dynamics simulation is tracked very well with both algorithms. There are again time delays to the reference signal that are larger when using the washout algorithm, due to the nature of the implemented filters. The roll velocity error is also

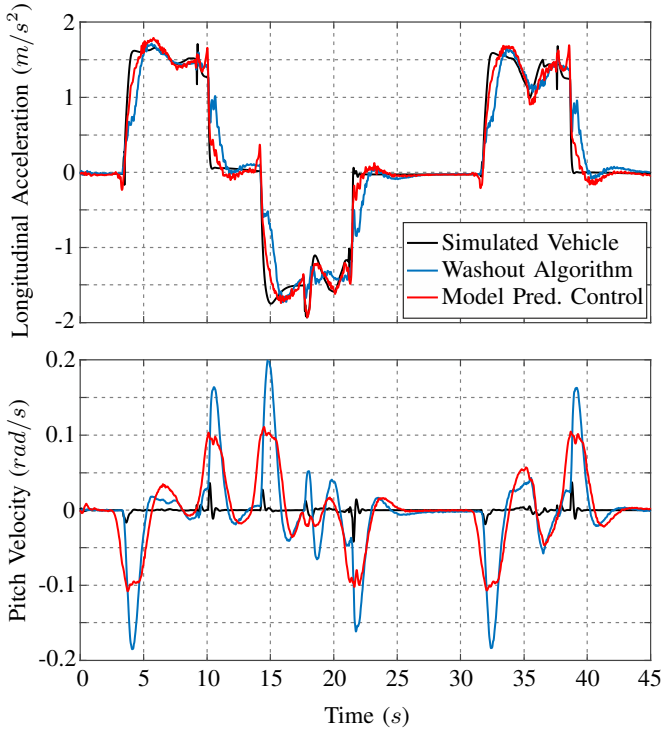


Figure 9. Longitudinal Acceleration and Pitch Velocity Tracking.

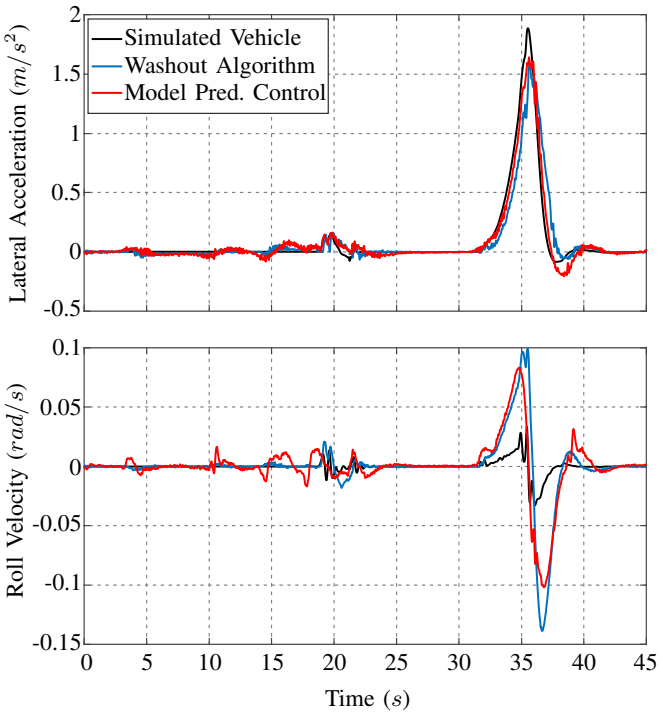


Figure 10. Lateral Acceleration and Roll Velocity Tracking.

larger than that of the MPC, even if the difference is smaller than in case of the pitch velocity. Thus, as a consequence for the interactive driving simulation, the resulting driving experience can be expected to be more realistic using the predictive control strategy, since smaller rotation rate errors are more difficult to detect for the human perception system. The vertical acceleration measured in the regarded driving scenario is illustrated in Figure 11. Here, it is noticeable that undesired vertical displacements occur due to the coupled degrees of freedom of the motion system, which cannot be compensated by any of the two approaches. However, these unpreventable errors are significantly lower and mostly below the perception threshold of the otoliths in the use of the predictive MCA, which can be explained by a more efficient coordination of the shaker and the motion platform. Regarding the examined driving maneuver, the application of the quality criteria introduced in the previous section results in the performance indicators $\lambda_{1,WO} = 1.74$ and $\lambda_{2,WO} = 0.92$ for the washout algorithm and $\lambda_{1,MPC} = 1.20$ and $\lambda_{2,WO} = 0.53$ for the predictive controller. This confirms the impression that the MPC-based algorithm achieves a higher control quality, which is primarily explained by the lower angular velocity and

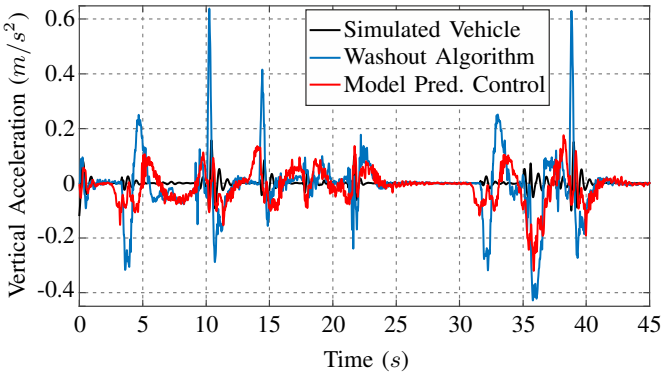


Figure 11. Vertical Acceleration Tracking.

 ISBN: 978-1-61208-756-6

vertical acceleration errors caused by the specific kinematics of the ATMOS driving simulator.

Figure 12 combines the evaluation of all nine test maneuvers in a common radar chart. The analysis of this graphic clearly shows the advantages of the optimization-based MCA in comparison to the washout algorithm, since smaller performance indicators are achieved in each scenario. It is noticeable that the perceived control quality, expressed by the indicator λ_2, yields small values close to zero when the MPC is used and therefore a good subjective driving impression can be expected. As already discussed in detail for driving maneuver "turning at a junction", these results can be explained with the angular velocity and vertical acceleration errors due to the coupled degrees of freedom, because of which an adequate reproduction of the simulated vehicles Motion Cues is a challenging task. Here, it is a great advantage of the MPC that the specific simulator kinematics are directly considered via existing model knowledge in the optimization algorithm. This allows undesired interactions to be taken into account in the planning of the motion trajectory and optimally compensated according to the current driving situation, which is a major benefit for interactive driving simulation.

VI. Conclusion and Future Work

In this paper, the development of a novel filter-based Motion Cueing Algorithm was presented. Motivated by the regarded hybrid kinematics motion system, the proposed algorithm features a dynamic position washout to any point within the simulator workspace, as well as a tilt coordination distribution strategy in order to make full use of the motion capabilities. Furthermore, this MCA was compared to a MPC-based algorithm by means of defined quality criteria and standard driving scenarios from the automotive industry. The objective evaluation of both Motion Cueing strategies proved a satisfactory control quality. However, due to the integration of model knowledge, the predictive MCA exhibits less control errors in angular velocities and vertical acceleration. For this reason, it is assumed that the subjective driving impression is more realistic when using the MPC, which is why this approach offers great potential for interactive driving simulation. On the other hand, the filter-based MCA has the advantages of simple implementation, good traceability and low computational effort, which relativizes the worse control quality in comparison to the optimization-based algorithm.

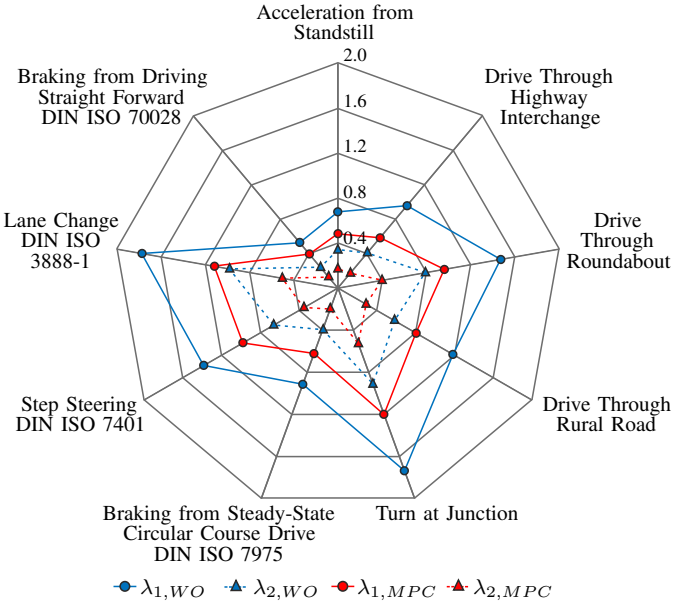


Figure 12. Evaluation of the Analyzed Test Maneuvers.

The future work will deal with the subjective validation of our observations. In this context, appropriate subject studies will be conducted in order to rate the resulting degree of immersion by human drivers. Besides, current research concentrates on the prediction of the future driver behavior and the associated reference trajectory for the model predictive MCA. Since the human driving behavior is predictable within certain limitations, a virtual driver model is applied to estimate the driver inputs within the time-limited prediction horizon at runtime.

References

[1] N. Rüddenklau, P. Biemelt, S. Henning, S. Gausemeier, and A. Trächtler, "Real-Time Lighting of High-Definition Headlamps for Night Driving Simulation," International Journal On Advances in Systems and Measurements, vol. 12, 2019, pp. 72–88.

[2] V. Melcher, S. Rauh, F. Diederichs, H. Widlroither, and W. Bauer, "Take-Over Requests for automated driving," Procedia Manufacturing, vol. 3, 2015, pp. 2867–2873.

[3] H. Bellem et al., "Can We Study Autonomous Driving Comfort in Moving-Base Driving Simulators? A Validation Study," Human Factors, vol. 59, no. 3, 2017, pp. 442–456.

[4] S. F. Schmidt and B. Conrad, "Motion Drive Signals for Piloted Flight Simulators," I Contract Report NASA, CR-1601, 1970.

[5] L. D. Reid and M. A. Nahon, "Flight Simulation Motion-Base Drive Algorithms: Part 1 - Developing and Testing the Equations," University of Toronto, UTIAS Report 296, 1985.

[6] T. Sammet, "Motion-Cueing-Algorithmen für die Fahrsimulation (Motion Cueing Algorithms for Driving Simulation)," Fortschritt Berichte-VDI Reihe 12 Verkehrstechnik/Fahrzeugtechnik, vol. 643, 2007.

[7] D. Stewart, "A Platform with Six Degrees of Freedom," Proceedings of the Institution of Mechanical Engineers, vol. 180, no. 1, 1965, pp. 371–386.

[8] D. Ariel and R. Sivan, "False Cue Reduction in Moving Flight Simulators," IEEE Transactions on Systems, Man, and Cybernetics, no. 4, 1984, pp. 665–671.

[9] M. Bruschetta, F. Maran, and A. Beghi, "A fast implementation of MPC-based motion cueing algorithms for mid-size road vehicle motion simulators." Vehicle System Dynamics, vol. 55, no. 6, 2017, pp. 802–826.

[10] J. O. Pitz, "Vorausschauender Motion-Cueing-Algorithmus für den Stuttgarter Fahrsimulator (Predictive Motion Cueing Algorithm for the Stuttgart Driving Simulator)," PhD Thesis, Universität Stuttgart, 2017.

[11] P. Biemelt, S. Henning, N. Rüddenklau, S. Gausemeier, and A. Trächtler, "A Model Predictive Motion Cueing Strategy for a 5-Degree-of-Dreedom Driving Simulator with Hybrid Kinematics," Driving Simulation Conference Europe 2018 VR (DSC), 2018, pp. 79–85.

[12] N. A. Pouliot, C. M. Gosselin, and M. A. Nahon, "Motion Simulation Capabilities of Three-Degree-of-Freedom Flight Simulators," Journal of Aircraft, vol. 35, no. 1, 1998, pp. 9–17.

[13] I. Al Qaisi and A. Trächtler, "Human in the Loop: Optimal Control of Driving Simulators and New Motion Quality Criterion," in 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC). IEEE, 2012, pp. 2235–2240.

[14] P. Grant, M. Blommer, B. Artz, and J. Greenberg, "Analysing Classes of Motion Drive Algorithms Based on Paired Comparison Techniques," Vehicle System Dynamics, vol. 47, no. 9, 2009, pp. 1075–1093.

[15] L. R. Young and J. L. Meiry, "A Revised Dynamic Otolith Model," Aerospace Medicine, vol. 39, no. 6, 1968, pp. 606–608.

[16] C. Fernandez and J. M. Goldberg, "Physiology of Peripheral Neurons Innervating Semicircular Canals of the Squirrel Monkey. II. Response to Sinusoidal Stimulation and Dynamics of Peripheral Vestibular System." Journal of Neurophysiology, vol. 34, no. 4, 1971, pp. 661–675.

 ISBN: 978-1-61208-756-6

Train Timetable Optimization for Parallel Single-track Sections During Track Closure

Akio Hada
Signalling and Transport Information Technology Division
Railway Technical Research Institute
Tokyo, Japan
e-mail: hada.akio.71@rtri.or.jp

Teodor Gradinariu
Rail System Department
International Union of Railways
Paris, France
e-mail: gradinariu@uic.org

***Abstract*—In parallel single-track sections, which are common in high-speed railways in Europe, when one of the tracks is closed due to vehicle failure or maintenance work, the other track can be used for travel in both directions to avoid significant delays or cancellation of train operation. However, for bidirectional operation on some sections of a parallel single-track line, it is necessary to adjust the original train timetable; usually, this adjustment causes some trains to be delayed. Therefore, in this study, we propose a mathematical model for train timetable planning when performing bidirectional operations on some parallel single-track sections to minimize the total delay of each train.**

Keywords-Parallel single-track; Track closure; Train timetable; Maintenance; Optimization.

I. INTRODUCTION

In a railway operation site, there are cases in which tracks for some sections have to be closed due to vehicle or signal failure. In addition, there are cases where it is necessary to close a track for maintenance work. If a track closure occurs during business hours, it might have a severe negative impact on train operation, and in some circumstances, it may become necessary to cancel train operations all over. However, on a parallel single-track, in which single tracks capable of bidirectional operation are laid in parallel, even if one track is closed, it is possible to avoid significant train delays and cancellation of trains. For example, tracks of many European high-speed railways such as Train a Grande Vitesse (TGV) [1] and InterCity Express (ICE) [2] have parallel single-track sections; a cross section is provided approximately every 50 km. Therefore, even if one of the tracks is closed, bidirectional operation can be performed on other tracks.

When performing two-way operation on some sections of a parallel single-track line, it is usually necessary to adjust the normal train operation timetable (hereafter, simply referred to as the original timetable). During this time, if the delay of each train increases due to changes in the travel time, it disturbs passengers, respectively as well as rolling stock schedule and crew schedule. Therefore, it is desirable that the travel time adjustment, when performing bidirectional operation, is planned well to minimize the delay.

Since track closure has significant impact on train operation, several studies for managing or planning a timetable during track closure have been carried out to date [3]-[9]. For example, [8] proposed the methodology to plan timetables in consideration of passenger services during track closure and [9] proposed the methodology to plan simultaneously the period of track closure and traffic flow. In these studies, it is acceptable to cancel train services during track closure. However, in practice, it may be required not to cancel train service and to plan a timetable sustaining the capacity in the original timetable during track closure. In this study, we propose a mathematical model for planning train timetable when performing bidirectional operation due to track closure to minimize total delay of each train without cancelling train services. We carry out simulations to verify the usefulness of the proposed mathematical model.

This paper is organized as follows. Section II explains the problem definition of train timetable planning during track closure. Section III presents a mathematical formulation for the problem. Section IV provide simulation results for several track closure periods. We conclude the paper in Section V.

II. TRAIN TIMETABLE PLANNING DURING TRACK CLOSURE

This section describes a problem definition of a train scheduling problem during track closure.

A. Bidirectional Operation on a Parallel Single-track Section

In parallel single-track sections capable of bidirectional operation, normal operation is carried out on two independent train routes with trains having different traveling directions. However, when a certain section of one route is closed, trains of that route are moved to the other route via a crossing line just before the closed section; they travel on that route and move to their original route via a crossing line at the end of the track closure. That is, when a certain section of one route is closed, two-way operation on the other route should be planned. Figure 1 shows an example of two independent

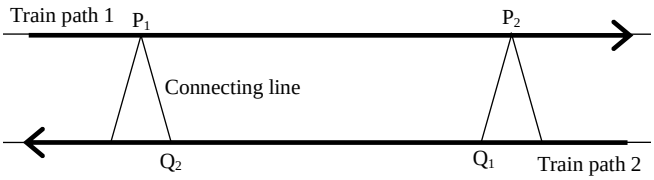


Figure 1. Train paths on a parallel single-track section

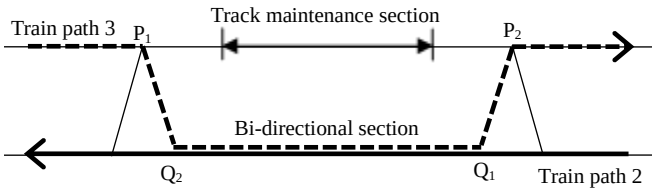


Figure 2. Change of the route of train path 1 due to track maintenance resulting in track closure

routes on a parallel single-track section, showing route 1 from left to right and route 2 from right to left in the case of normal operation. Further, P_1 and P_2 indicate the connection points of a crossing line on route 1, and Q_1 and Q_2 the connection points of a crossing line on route 2. Figure 2 is an example of a case where a track section between point P_1 and point P_2 is closed due to maintenance work; bidirectional operation between point Q_1 and point Q_2 is also shown as route 1 in Figure 1, which is then changed to route 3 in Figure 2 (trains on route 3 travel in the order $P_1 \rightarrow Q_2 \rightarrow Q_1 \rightarrow P_2$).

It is impossible for two trains with different traveling directions to simultaneously travel on a section on which bidirectional operation can be performed (hereafter, simply referred to as a bidirectional section). In other words, trains with the same travelling direction can continue running on a bidirectional section; however, if two trains with different traveling directions travel on a bidirectional section, train on one route needs to wait for the train on the other route to pass. Therefore, it is necessary for the train timetable on a bidirectional section to be planned by considering the trains with different traveling directions.

B. Train Scheduling Problem During Track Closure

There is a concern that passenger dissatisfaction will increase due to an increase in the congestion rate of trains due to longer operation intervals if trains are cancelled when performing a bidirectional operation. In addition, if the order of trains is changed, it will interfere with the vehicle operation schedule and the crew schedule. Moreover, if the operation time of each train is advanced, passengers may not be able to get on the scheduled trains. Therefore, the following assumptions with respect to a train schedule are made when bidirectional operation is performed due to track closure.

1. Trains will not be canceled.
2. The train operation sequence will be maintained as per the original timetable.
3. The operation time of each train will be the same as or later than the original timetable.

Further, the following assumptions are made with respect to track closure section and period (start time to end time of track closure).

4. Only one track is closed on a certain section.
5. The track closure section is between the crossing that is before the section closed for maintenance and the next crossing (section between points P_1 and P_2 in Figure 2).
6. A train will not be in the track closure section during the track closure period.

From assumption 6, for a route with a closed track section, a train traveling on the track closure section during the track closure period will be moved to the other track in the bidirectional section. Then, the following assumptions are made for routes 1, 2, and 3 in Figures 1 and 2.

7. Trains traveling on the track closure section on route 1 during the track closure period as per the original timetable travel on route 3 until the track closure ends and on route 1 after the track closure ends.
8. Trains traveling on the track closure section on route 2 during the track closure period as per the original timetable always travel on the same route.

Moreover, to smoothing the delay time from the original timetable of each train, it is necessary to plan such that the delay of each train is within a certain limit. Then, the following assumption is made.

9. The delay time of each train has an upper limit.

The problem to be addressed in this study is to plan the train timetable when performing bidirectional operation during track closure to minimize the total delay of each train under the above conditions. For this, the following times, shown in Figure 2, need to be planned.

- The time when trains traveling on the track closure section on route 1 during the track closure period as per the original timetable pass point P_1
- The time when trains traveling on the track closure section on route 2 during the track closure period as per the original timetable pass point Q_1

As described above, the timetable of the bidirectional section must be planned considering train schedules of routes with different directions simultaneously. On the contrary, a train timetable, not on a bidirectional section, can independently adjust train schedules of one-directional routes. Therefore, we assume that, after planning a train schedule for a bidirectional section, the timetable for other sections can be adjusted for each route for synchronization. In the following, we also assume that the track closure section and track closure period are given.

III. FORMULATION OF A TRAIN TIMETABLE PLANNING PROBLEM

This section describes a mathematical formulation of a train timetable planning problem defined in Section II.

A. Terminology

The following terms are defined with respect to Figures. 1 and 2. For an original timetable of route 1, let $H_1 = \{1,2,\ldots,\mu\}$ be the set of all trains traveling on the track closure section (between points P_1 and P_2) during the track closure period. For an original timetable of route 2, let $H_2 = \{\mu+1, \mu+2, \ldots, \mu+\nu\}$ be the set of all trains traveling on the bidirectional section (between points Q_1 and Q_2) during the track closure

period. We also let $H = H_1 \cup H_2$. Note that the operation sequence, according to the original timetable, of trains included in H_1 and H_2 is $1 \rightarrow 2 \rightarrow \ldots \rightarrow \mu$ and $\mu + 1 \rightarrow \mu + 2 \rightarrow \ldots \rightarrow \mu + \nu$, respectively. For example, the track closure period is assumed to be from 13:10 to 15:10 and the time required for trains on route 1 to travel on the track closure section and the time required for trains on route 2 to travel on the bidirectional section is 1 hour. In this case, if trains passing through point P_1 at the following times exist in the original timetable of route 1, then trains passing point P_1 at the times indicated by the solid underline are included in H_1.

11:30, 12:00, 12:30, 13:00, 13:30, 14:00, 14:30, 15:00, 15:30, 16:00

Moreover, if trains passing through point Q_1 at the following times exist in the original timetable of route 2, then trains passing point Q_1 at the times indicated by the dotted underline are included in H_2.

11:35, 12:05, 12:35, 13:05, 13:35, 14:05, 14:35, 15:05, 15:35, 16:05

It should be noted that trains passing through point P_1 (respectively, point Q_1) before the start time of the track closure period are also included in H_1 (respectively, H_2).

In Section II, when the track between points P_1 and P_2 of route 1 is closed, we set up a problem to determine the times for passing through point P_1 for trains included in H_1 and times for passing through point Q_1 for trains included in H_2 to minimize the total delay of all trains. In the following, we formulate this problem into a mathematical programming problem. Here, from assumptions 7 and 8 in Section II, trains in H_1 are scheduled to travel on route 3 until the end time of track closure and are scheduled to travel on route 1 after the end time of track closure. On the other hand, trains in H_2 are always scheduled to travel on route 2.

The following times defined below are assumed to be given.

ϖ_η: The time required for train $\eta \in H_1$ to travel from point P_1 to point P_2 on route 3 or for train $\eta \in H_2$ to travel from point Q_1 to point Q_2 of route 2 (traveling time)

ω_η: The time required for switching the travel route from route 3 to route 2 for train $\eta \in H_1$ and from route 2 to route 3 for train $\eta \in H_2$ (route switching time)

α_η: The time interval from the time train $\eta\ (\neq \mu) \in H_1$ passes through point P_1 to the time when the following train $\eta + 1 \in H_1$ passes through point P_1 or from the time when train $\eta\ (\neq \mu + \nu) \in H_2$ passes through point Q_2 to the time when the following train $\eta + 1 \in H_2$ passes through point Q_2 (minimum headway)

δ_η: The upper limit of the permissible delay time of train $\eta \in H$ (permissible delay time)

Here, it is assumed that the times defined above are in minutes and are all non-zero. In addition, it is assumed that traveling time ϖ_η is defined considering the speed regulation in a section including switches, reduced speed operation on a maintenance section, etc., and that minimum headway α_η is defined considering the running speed of each train, various restrictions on train operation, etc.

In the original timetable of trains included in H, train 1 passes through point P_1 first, and train $\mu + 1$ passes through point Q_1 first. If we let σ' be the first time when train 1 passes point P_1 and the time when train $\mu + 1$ passes through point Q_1 in the original timetable, we only need to take time σ' and later into consideration in our problem. Therefore, we define $T_1 = \{1,2,\ldots,\kappa\}$ as the set of times with one-minute intervals from σ' to the end time of the track closure. Here, the first element of T_1 represents time σ' and the κth element represents the end time of track closure. For example, if time σ' is 12:00 and the end time of the track closure is 15:00, then the elements 1, 2 and 3 of T_1 will be 12:00, 12:01, and 12:02, respectively, and element $\kappa = 181$ of T_1 indicates the end time of track closure, i.e., 15:00. We also define the following.

σ_η: The time in T_1 when train $\eta \in H_1$ passes through point P_1 in the original timetable of route 1 or when train $\eta \in H_2$ passes point Q_1 in the original timetable of route 2.

For example, if σ' is 12:00 and the time when train $\eta \in H_1$ passes through point P_1 in the original timetable is 13:10, then $\sigma_\eta = 71$.

In the original timetable of trains included in H, train μ passes through point P_1 last and train $\mu + \nu$ passes through point Q_1 last. On the other hand, from assumption 9 in Section II, the delay of train $\eta \in H$ must be less than or equal to δ_η. Therefore, if we let τ' (note that it is the time in T_1) to be the later time between $\sigma_\mu + \delta_\mu$ and $\sigma_{\mu+\nu} + \delta_{\mu+\nu}$, in addition to T_1 previously defined, we only need to take the time interval from $\kappa+1$ to τ' into consideration for our problem. However, depending on the problem, there exist cases where time τ' is earlier than time $\kappa+1$. Therefore, we also set $\lambda = \max\{\tau', \kappa+1\}$. Moreover, let $T_2 = \{\kappa+1, \kappa+2, \ldots, \lambda\}$ be the set of times of one-minute intervals from time $\kappa+1$ to λ, and $T = T_1 \cup T_2$.

B. Definition of the Train Scheduling Problem

Let $A = H \times T$ be the set of direct products of the set of trains H and the set of times T; the subsets of A that restrict the set of train H to H_1 and H_2 are defined as follows.

$$A_1 = \{(\eta, \tau) \in A \mid \eta \in H_1,\ \tau \in T\}$$

$$A_2 = \{(\eta, \tau) \in A \mid \eta \in H_2,\ \tau \in T\}$$

Furthermore, variable $\xi_{\eta\tau}$ is assigned to element (η, τ) of A. Variables $\xi_{\eta\tau}$, $(\eta, \tau) \in A$ is 1 if the train $\eta \in H_1$ ($\eta \in H_2$) passes through point P_1 (Q_1) at time $\tau \in T$ and is 0 otherwise. Here, the total number of variables $\xi_{\eta\tau}$ is $|A| = (\mu + \nu) \times \lambda$.

As mentioned previously, trains in the original timetable were assumed to not be canceled. Therefore, train $\eta \in H$ must be planned to pass through point P_1 or point Q_1 at any time

 ISBN: 978-1-61208-756-6

between 1 and λ. That is, variables $\xi_{\eta\tau}$ must satisfy the following.

$$\sum_{1 \le \tau \le \lambda} \xi_{\eta\tau} = 1, \ \eta \in H \quad (1)$$

In addition, it was assumed that the operation time of each train must be the same as or later than the original operation time and that the delay of each train should not exceed the permissible delay. Therefore, train $\eta \in H$ must be planned to pass through point P_1 or point Q_1 at any time from σ_η to $\sigma_\eta + \delta_\eta$. That is, let $\beta_\eta = \sigma_\eta + \delta_\eta$, $\eta \in H$, variables $\xi_{\eta\tau}$ must satisfy the following.

$$\sum_{\sigma_\eta \le \tau \le \beta_\eta} \xi_{\eta\tau} = 1, \ \eta \in H \quad (2)$$

Furthermore, the train operation sequence is assumed to be maintained as per the original timetable. Therefore, all trains operated before train $\eta \in H_1 (\eta \ne 1)$ must pass through point P_1 before time τ in order for train η to pass through point P_1 at time τ. Similarly, all trains operated before train $\eta \in H_2 (\eta \ne \mu + 1)$ must pass through point Q_1 before time τ for train η to pass through point Q_1 at time τ. Here, for the first train in H_1 and first train $\mu + 1$ in H_2, there is no need to consider the above constraints. Moreover, because only those trains can pass through point P_1 or point Q_1 at time 1, it is not necessary for trains passing through point P_1 or point Q_1 to consider the above constraints. Now, we define the subset of A_1 obtained by subtracting the elements for the case $\eta = 1$ and the case $\tau = 1$ and the subset of A_2 obtained by subtracting the elements for the case $\eta = \mu + 1$ and the case $\tau = 1$ as follows.

$$A'_1 = \{(\eta, \tau) \in A_1 \mid \eta \ne 1, \ \tau \ne 1\}$$
$$A'_2 = \{(\eta, \tau) \in A_2 \mid \eta \ne \mu + 1, \ \tau \ne 1\}$$

Then, the constraint condition on the train operation sequence is expressed as follows.

$$\sum_{1 \le \tau' < \tau} \xi_{\eta'\tau'} \ge \xi_{\eta\tau}, \quad 1 \le \eta' < \eta, \ (\eta, \tau) \in A'_1 \quad (3)$$

$$\sum_{1 \le \tau' < \tau} \xi_{\eta'\tau'} \ge \xi_{\eta\tau}, \quad \mu + 1 \le \eta' < \eta, \ (\eta, \tau) \in A'_2 \quad (4)$$

From the definition of A'_1, A'_2 for any $(\eta, \tau) \in A'_1$ in (3) and any $(\eta, \tau) \in A'_2$ in (4), there exists η' where $1 \le \eta' < \eta$ and η' where $1 \le \eta' < \eta$, respectively. Figure 3 shows an example in which $H_1 = \{1,2,3,4\}$, $T_1 = \{1,2,\ldots,10\}$, and $T_2 = \{11,12,\ldots,15\}$, and the element of row η and column τ represents variable $\xi_{\eta\tau}$. Note that the rows corresponding to H_2 are omitted. In addition, the gray parts indicate variables $\xi_{\eta\tau}$, $(\eta, \tau) \in A'_1$. In Figure 3, in the case in which train 3 passes point P_1 at time 5 is when $\xi_{35} = 1$; the variables surrounding the bold frame must satisfy $\xi_{11} + \xi_{12} + \xi_{13} + \xi_{14} = 1$ and $\xi_{21} + \xi_{22} + \xi_{23} + \xi_{24} = 1$.

		T_1										T_2				
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
H_1	1	ξ_{11}	ξ_{12}	ξ_{13}	ξ_{14}											
	2	ξ_{21}	ξ_{22}	ξ_{23}	ξ_{24}											
	3					$\xi_{35} = 1$										
	4															

Figure 3. Constraints on train operation sequences

Next, we consider the headway between trains. From the definition of minimum headway α_η, when train $\eta (\ne \mu) \in H_1$ passes through point P_1 at time $\tau \in T$, the following train $\eta + 1 \in H_1$ cannot pass through point P_1 before time $\tau + \alpha_\eta$. Similarly, when train $\eta (\ne \mu + \nu) \in H_2$ passes through point Q_1 at time $\tau \in T$, the following train $\eta + 1 \in H_2$ cannot pass through point Q_1 before time $\tau + \alpha_\eta$. Note that the last train μ in H_1 and the last train $\mu + \nu$ in H_2 are not included in the problem. We define a subset of A where the elements in case of $\eta = \mu$ and $\eta = \mu + \nu$ are excluded as follows.

$$A'' = \{(\eta, \tau) \in A \mid \eta \ne \mu, \ \mu + \nu\}$$

Moreover, depending on time τ and minimum headway α_η, $\tau + \alpha_\eta$ may exceed the last time λ in T; we introduce the following:

$$\alpha(\eta, \tau) = \min\{\tau + \alpha_\eta, \lambda + 1\}, \quad (\eta, \tau) \in A''$$

Then, the constraint conditions on the operation time interval are expressed as follows. In (5), from the definition of A'' and $\alpha_\eta > 0$, there always exist $\eta + 1$ and τ' satisfying $\tau \le \tau' < \alpha(\eta, \tau)$ for an arbitrary $(\eta, \tau) \in A''$.

$$\sum_{\tau \le \tau' < \alpha(\eta, \tau)} \xi_{\eta+1, \tau'} + \xi_{\eta\tau} \le 1, \quad (\eta, \tau) \in A'' \quad (5)$$

As mentioned above, trains with different traveling routes cannot travel on a bidirectional section at the same time. Therefore, in cases that train $\eta' \in H_2$ travels after train $\eta \in H_1$ or that train $\eta' \in H_1$ travels after train $\eta \in H_2$ on a bidirectional section, train η' can travel on the bi-directional section only after a train η passes through the bidirectional section. Note that trains included in H_1 and H_2 travel on a bidirectional section until the track closure time ends. Thus, let A_1 and A_2 be limited to the time until track closure ends, and set $A'''_1 = H_1 \times T_1$ and $A'''_2 = H_2 \times T_1$. The time required for train $\eta \in H$ to travel on a bidirectional section is ϖ_η and the time required for switching the travel route after train η passes through the bidirectional section is ω_η. Therefore, if train $\eta \in H_1$ passes through point P_1 at time $\tau \in T_1$, train $\eta' \in H_2$ can pass through point Q_1 only after time $\tau + \varpi_\eta + \omega_\eta$. Similarly, if train $\eta \in H_2$ passes through point Q_1 at time $\tau \in T_1$, train $\eta' \in H_1$ can pass through point P_1 only after time $\tau + \varpi_\eta + \omega_\eta$. Note that depending on traveling time ϖ_η and route switching time ω_η, $\tau + \varpi_\eta + \omega_\eta$ may exceed the last time λ in T; we introduce the following:

$$\varepsilon(\eta, \tau) = \min\{\tau + \varpi_\eta + \omega_\eta, \lambda + 1\}, \quad (\eta, \tau) \in A'''_1$$

On the other hand, from assumption 7 in Section II, trains included in H_1 travel on original route 1 as soon as the track closure ends. Therefore, when train $\eta \in H_2$ travels on the bidirectional section, we need to consider the constraint conditions for train $\eta' \in H_1$ mentioned above until time κ when the track closure ends; we do not need to consider it after time $\kappa + 1$. Here, we introduce the following.

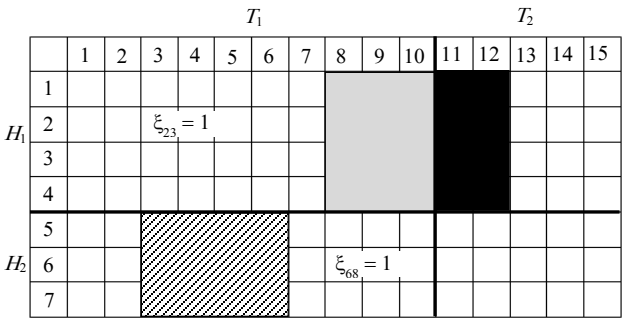


Figure 4. Constraints on switching train paths

$$\phi(\eta, \tau) = \min\{\ \tau + \varpi_\eta + \omega_\eta,\ \kappa + 1\}, \quad (\eta, \tau) \in A'''_2$$

Then, the constraint conditions for preventing the trains included in H_1 and H_2 from simultaneously traveling on the bidirectional section are expressed as follows.

$$\sum_{\tau \le \tau' < \varepsilon(\eta, \tau)} \xi_{\eta'\tau'} + \xi_{\eta\tau} \le 1, \quad \eta' \in H_2, (\eta, \tau) \in A'''_1 \qquad (6)$$

$$\sum_{\tau \le \tau' < \phi(\eta, \tau)} \xi_{\eta'\tau'} + \xi_{\eta\tau} \le 1, \quad \eta' \in H_1, (\eta, \tau) \in A'''_2 \qquad (7)$$

From $\varpi_\eta, \omega_\eta > 0$, there always exist τ' where $\tau \le \tau' < \varepsilon(\eta, \tau)$ for any $(\eta, \tau) \in A'''_1$ in (6), and τ' where $\tau \le \tau' < \phi(\eta, \tau)$ for any $(\eta, \tau) \in A'''_2$ in (7). Figure 4 shows an example in which $H_2 = \{5,6,7\}$ is added to Figure 3. If we set $\xi_{23} = 1$ when $\varpi_2 = 3$ and $\omega_2 = 1$ in Figure 4, equation (6) shows that all shaded variables must be set to 0. That is, when train 2 passes through point P_1 at time 3, it indicates that trains included in H_2 cannot pass through point Q_1 from time 3 to 6. In addition, if we set $\xi_{68} = 1$ when $\varpi_6 = 4$ and $\omega_6 = 1$ in Figure 4, equation (7) shows that the gray variables must be set to 0. Here, because the trains included in H_1 can travel on original route 1 as soon as the track closure ends, the constraint conditions given by (7) are not imposed for the black variables.

Based on this, the problem defined in this section is formulated into the following mathematical programming problem (P).

(P) Minimize $\sum_{\eta \in H} \sum_{\tau \in T} (\tau - \sigma_\eta) \xi_{\eta\tau}$ (8)

Subject to: Equations (1) - (7)

$$\xi_{\eta\tau} \in \{0, 1\}, \quad (\eta, \tau) \in A \qquad (9)$$

In this problem (P), equation (8) is the objective function to minimize the sum of the delay of each train. Equation (9) is the binary condition of the decision variable. Here, the problem to minimize the maximum delay of each train, not the total delay time of each train, can be formulated as the following mathematical programming problem (Q) by introducing the variable ψ (≥ 0).

(Q) Minimize ψ

Subject to: Equations (1) - (7), (9)

$$\sum_{\tau \in T} (\tau - \sigma_\eta) \xi_{\eta\tau} \le \psi, \quad \eta \in H \qquad (10)$$

For the problem (P), there may not exist any feasible solution. However, the absence of a feasible solution shows that it is impossible to adjust the original timetable under the given parameter settings. In other words, in this case, it shows that measures such as extending the permissible delay time, shortening the minimum headway, or changing the period track closure are necessary. For train operation during track closure, it is also important to determine whether a feasible solution exists for the problem (P) for a given parameter setting. On the other hand, recent mathematical programming software are becoming faster every year. It is possible to determine the existence of a feasible solution and the calculation of an optimal solution for the problem with actual scale of parameters in relatively short time. Therefore, it is possible to use mathematical programming software such as Gurobi or MATLAB to determine the existence of a feasible solution and the calculation of an optimal solution for problems (P) and (Q).

IV. SIMULATIONS

In this section, we describe the results of a simulation executed to examine the variation in total delay time when the track closure period are 3 hours, 6 hours and 9 hours. In the simulations to be described, we set the parameters as follows.

- The unit time is one minute.
- Trains on the route 1 pass point P_1 at the time 7:31, 8:01, 9:01, 9:31, 10:31, 11:31, 12:31, 13:01, 13:31, 15:01, 15:31, 16:31, 17:31, 18:31, 19:01, 19:31, 20:31, 21:31 as per the original timetable.
- Trains on the route 2 pass point Q_1 at the time 5:59, 7:20, 8:14, 8:50, 9:43, 10:43, 11:50, 12:20, 12:43, 13:50, 14:41, 15:50, 16:20, 16:50, 17:50, 18:20, 18:50, 19:20, 20:20 as per the original timetable.
- We set the input parameters $\varpi_\eta = 20,\ \eta \in H,\ \omega_\eta = 1,\ \eta \in H,\ \alpha_\eta = 2,\ \eta \in H - \{\mu, \mu + \nu\},\ \delta_\eta = 30,\ \eta \in H$ respectively.

Table I, Table II and Table III show the results of the track closure periods 3 hours, 6 hours and 9 hours, respectively. For Table I, Table II and Table III to be presented later the following should be noted.

- ω_{start} is the start time of track closure.
- ω_{end} is the end time of track closure.
- $|H_1|$ and $|H_2|$ are the number of trains in H_1 and H_2, respectively.
- $|T|$ is the number of elements in T.
- δ_{total} is the optimal solution value of the problem (P).
- δ_{ave} is the average delay time defined by $\delta_{ave} = \delta_{total} / |H|$.
- *time* is the computational time to judge the existence of a feasible solution of the problem (P) or calculate the optimal solution of the problem (P).
- We implemented simulations using Gurobi Optimizer 8.1.0 on a personal computer having the following specification, Intel® Core™ i7-6700 CPU 3.40GHz, 8.00GB RAM, running Windows 10 Pro.

From Tables I to III, the average delay time for track closure period of 3 hours, 6 hours and 9 hours are respectively 6.24s, 6.38s and 6.32s and there are not significant differences between those. In general, it is more economical for track

 ISBN: 978-1-61208-756-6

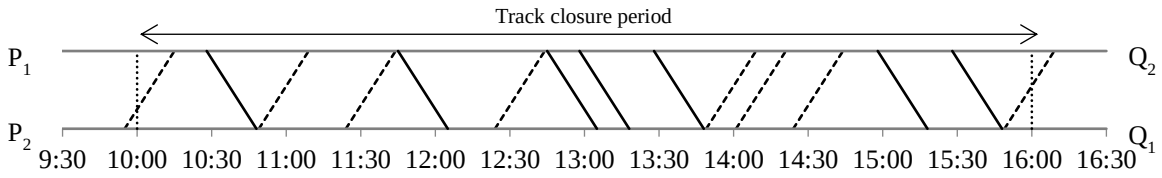


Figure 5. Diagrams for the adjusted train timetable calculated by the proposed methodology

maintenance work to perform a few long-term work rather than many short-term work. Therefore, in the original timetable in this simulation, it is considered to be more economical to plan long-term work rather than short-term work. In this way, the mathematical model proposed in this paper can be used to find a cost-effective maintenance plan. Figure 5 shows the results calculated by the proposed mathematical model in the case track closure period 10:00-16:00 in Table II. In Figure 5, the vertical and horizontal axes means distance (or train position) and time, respectively, diagrams of trains included in H_1 and H_2 are respectively shown by solid lines and dotted lines and the points P_1, P_2, Q_1 and Q_2 have the same meaning as in Figure 1 and Figure 2.

TABLE I. SIMULATION RESULTS FOR TRACK CLOSURE PERIOD OF 3 HOURS

ω_{start}	ω_{end}	$\|H_1\|$	$\|H_2\|$	$\|T\|$	δ_{total}	δ_{ave}	*time*
6:00	9:00	3	1	120	21	5.25	0.21
7:00	10:00	4	3	177	45	6.43	1.13
8:00	11:00	4	3	184	42	6.00	1.28
9:00	12:00	4	4	184	59	7.38	1.70
10:00	13:00	4	4	213	52	6.50	1.94
11:00	14:00	4	3	158	52	7.43	1.05
12:00	15:00	4	4	184	35	4.38	1.64
13:00	16:00	4	4	184	36	4.50	1.72
14:00	17:00	3	4	181	33	4.71	1.27
15:00	18:00	4	3	184	51	7.29	1.30
16:00	19:00	4	4	186	57	7.13	1.64
17:00	20:00	4	4	154	60	7.50	1.40
18:00	21:00	4	5	181	60	6.67	2.18

TABLE II. SIMULATION RESULTS FOR TRACK CLOSURE PERIOD OF 6 HOURS

ω_{start}	ω_{end}	$\|H_1\|$	$\|H_2\|$	$\|T\|$	δ_{total}	δ_{ave}	*time*
6:00	12:00	6	5	274	80	7.27	5.44
7:00	13:00	8	6	360	97	6.93	12.47
8:00	14:00	8	6	364	94	6.71	13.14
9:00	15:00	8	8	390	94	5.88	18.46
10:00	16:00	7	8	367	88	5.87	15.67
11:00	17:00	7	7	338	85	6.07	12.03
12:00	18:00	7	7	338	86	6.14	12.00
13:00	19:00	8	8	390	93	5.81	18.76
14:00	20:00	8	7	361	93	6.20	15.27
15:00	21:00	8	8	364	111	6.94	18.02

TABLE III. SIMULATION RESULTS FOR TRACK CLOSURE PERIOD OF 9 HOURS

ω_{start}	ω_{end}	$\|H_1\|$	$\|H_2\|$	$\|T\|$	δ_{total}	δ_{ave}	*time*
6:00	15:00	10	9	480	115	6.05	35.68
7:00	16:00	11	10	514	133	6.33	49.59
8:00	17:00	11	10	544	127	6.05	54.08
9:00	18:00	11	11	544	145	6.59	59.26
10:00	19:00	11	12	573	145	6.30	69.41
11:00	20:00	11	11	518	145	6.59	54.74
12:00	21:00	11	12	518	146	6.35	61.77

V. CONCLUSION

In this paper, we addressed the problem of how to adjust the original timetable for parallel single-track sections during track closure so as to minimize the sum of the delay of each train by formulating the task as a mathematical programming problem. Then, we verified the effectiveness of our method through simulations. However, there are still several problems that remain. For example, we need to construct a method to recover the delay caused by track closure after track closure ends. We also need to extend our method so that it can be used in the sections including multiple track closure sections.

REFERENCES

[1] French National Railway Company, https://www.sncf.com/fr [retrieved: September, 2019]

[2] Deutsche Bahn, https://www.bahn.de/p/view/index.shtml [retrieved: September, 2019]

[3] A. Albrecht, D. Panton, and D. Lee, "Rescheduling rail networks with maintenance disruptions using problem space search," Computers & Operations Research, vol. 40, no. 3, pp. 703-712, 2013.

[4] M. Lake and L. Ferreira, "Minimising the conflict between rail operations and infrastructure maintenance," In: Taylor, M. (Ed.), Transportation and Traffic Theory in the 21st Century: Proceedings of the 15th International Symposium on Transportation and Traffic Theory, Pp. Elsevier, Oxford, UK, pp. 63-80.

[5] G. Budai-Balke, "Operations research models for scheduling railway infrastructure maintenance (PhD thesis)," Erasmus University, Rotterdam, The Netherlands.

[6] F. Peng, S. Kang, X. Ouyang, K. Somani, and D. Acharya, "A Heuristic approach to the railroad track maintenance scheduling problem," Comput Aided Civil Infrastruct. E.g. vol. 26, no. 2, pp. 129-145, 2011.

[7] P. Vansteenwegen, T. Dewilde, S. Burggraeve, and D. Cattrysse, "An interative approach for reducing the impact of infrastructure maintenance on the performance of railway systems," European Journal of Operational Research, vol. 252, pp. 39-53, 2016.

[8] R. L. Burdett and E. Kozan, "Techniques for inserting additional trains into existing timetables," Transportation Research Part B, vol. 43, pp.821-836, 2009.

[9] M. Forsgren, M. Aronsson and S. Gestrelius, "Maintaining tracks and traffic flow at the same time," Journal of Rail Transport Planning & Management, vol. 3, pp. 111-123, 2013.

[10] Gurobi Optimizer, http://www.gurobi.com/ [retrieved: September, 20191

 ISBN: 978-1-61208-756-6

Cost Evaluation System for Plant Transportation Over Land

In-Hak Lee
Dept. Mechanical Engineering
Yonsei university
Seoul, Korea, Republic of
e-mail: inhak2001@yonsei.ac.kr

Se-Hyun Hwang
Dept. Mechanical Engineering
Yonsei university
Seoul, Korea, Republic of
e-mail: hwanghyun3@gmail.com

Ho-Joon Son
Dept. Mechanical Engineering
Yonsei university
Seoul, Korea, Republic of
e-mail: hojoonson@yonsei.ac.kr

Soo-Hong Lee
Dept. Mechanical Engineering
Yonsei university
Seoul, Korea, Republic of
e-mail: shlee@yonsei.ac.kr

***Abstract*— The purpose of this study is to develop a system for efficient cost calculation of transportation processes among many processes of plant construction. The developed system is called P-TAIS, and the transportation process can be analyzed by module and vehicle information through the system. The analysis includes in-path curve analysis, slope analysis, obstacle analysis, path width analysis, and vehicle analysis. The analysis then calculates the overall cost of the transportation process.**

Keywords-Modular plant; P-TAIS; Custom Path; Path assessment; SPMT; Cost optimize;

I. INTRODUCTION

As complex processes are required to build a plant [1],[5], careful decisions are required for the construction, and the estimation of the total cost before construction is very important [2].

There are two known methods to construct a plant [3]. The first method is stick-build, where materials are transported to the site, and then fabricated to become a part of the plant. The second method is modular method, where sections of a plant are pre-made and transported to the site to later be assembled. There exist pros and cons of each construction method, depending on the requirements of each plant. When both methods are feasible, however, the modular method is usually more advantageous in terms of construction period and cost [4].

This research focuses on the transportation of plant module on land. Compared to the stick build method, modules getting larger and heavier in modular plant for optimized cost, their transportation on land can act a constraint at the scale of entire plant project. Therefore, this research aims to provide a web service for cost assessment and integrated management on land transportation required for plant projects. This service is named Plant-Transportation Analysis Information System(P-TAIS) [6].

Section 2 introduces the structure of the P-TAIS system and the role of each page. Section 3 introduces the five analyzes used in the P-TAIS system. The brief method of each analysis and the factors for cost analysis are explained. Section 4 describes the cost analysis method using the factors obtained through the analysis. Finally, a short summary of the overall system and future work to improve the shortcomings of this system are presented.

II. THE P-TAIS SYSTEM

P-TAIS aims to simplify this process and replace it with a web-based analysis. It allows freedom of access to information for all the members participating in a plant construction project. All of the information on past projects can be managed in an organized way through P-TAIS, and for the current ones P-TAIS can be used for real-time sharing of information. The client only needs a web browser to use P-TAIS, and no data is stored on client.

A. P-TAIS Structure

The schematics of P-TAIS is shown in Figure 1. To reach the analysis page on the land transportation of a module, the user is asked to create/manage project and the modules that the project belong. A module can be transported through *Custom Path*, which is entered by specifying waypoints in addition to the origin and destination. Furthermore, there is also a link to swept-path analysis in the analysis page.

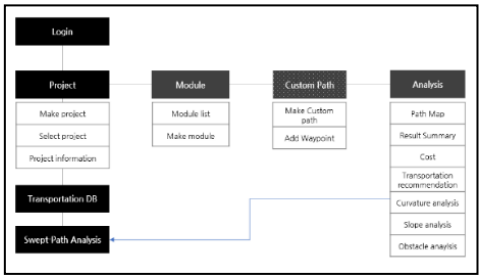


Figure 1. P-TAIS page structure

B. The Log-In Page

First, the company that is going to use P-TAIS is given username and password after an agreement. Whether P-TAIS will be a paid service has not been decided yet. If a client

 ISBN: 978-1-61208-756-6

requests a computing power more than we currently own, we can negotiate over the cost of setting up additional computing resources.

C. The Projects Page

To create a project, it requires for user to enter the name of the project and the area in which the land transportation will be carried on. Once the project area is set, the area will be used to display all the maps in the context of that project. The projects are sorted by the time of the last use, and the number of registered modules is noted.

D. The Modules Page

Each project needs the modules to be registered. On module creation, user assigns a number to the new module, defines the origin and destination, the weight, and the size of a module. Each module. At the moment when the origin and destination are both set, route(s), multiple if possible, are created by Google Maps API.

E. The Custom Path Page

In addition to the route generated on the module registration, users can create more routes by adding up to 8 waypoints. This functionality can be used, for instance, to satisfy the requirement of passing through a gas station.

F. The Trasporters Page

Users can register means of transport that are available for the company. The types of those can be either of flatbed truck or of Self-Propelled Modular Transporter (SPMT). The list of the registered transporters is also available for viewing. The analysis page will later offer a recommendation on the best mode of transport among various configuration possible with the given transporters. Required information for each transporter are size, payload, Center of Gravity (CoG).

III. ANALYSIS

The analysis will result in a cost assessment for each of the route, based on the project, module, transporter information registered beforehand. The analysis will cover the following aspect of a particular route: curvature of the road, slope of the road, obstacles along the route, width of the road, transporter.

A. Curvature of the Road

First of all, Google Maps directions API provides the nodes consisting the route. Given this raw data, the curvature of the road is defined as the change in the direction of travel along the route at each node. The maximum of this value is calculated along the route. The analysis page contains a shortcut to start a swept-path analysis at this node of maximum curvature.

B. Slope of the Road

There is a limit to which SPMT can compensate for a slope of the road. Depending on the model, a SPMT can keep its payload level, on a surface sloped up to 6-7 degrees. Beyond this slope is where the payload is no longer secure. In this case, a further structural analysis should be carried out on the behavior of the module. The difference in elevation between the nodes divided by distance between the nodes, is equal to the slope of the road.

C. Obstacles along the Route

Obstacles on the route of a transportation can prevent the module from passing through. Obstacles along the route are detected automatically by object classification algorithm based on deep learning called YOLO v3 [7]. YOLOv3 is faster than other deep learning algorithms, reducing the time it takes to deal with classification of simple obstacles. Users have to upload a driving video along the route in analysis page, which is then processed to yield a result of type, position, height of the obstacles in the video.

D. Width of the Road

If the width of the road at a certain point along the route is wider than the width of the SPMT combination, either the road has to be widened or different route has to be considered. As mentioned above, the width of the road is an important factor, which can be analyzed using the Pix2Pix algorithm, which is widely used for satellite image analysis [8]. Pix2Pix deep learning algorithm processes satellite images to automatically count the number of pixels making up the road and it can be changed width of the road.

E. Transporter

SPMTs can move in a formation to carry a payload that is beyond the capabilities of individual SPMT. The analysis page contains the recommendation of the configuration of SPMTs, if more than one SPMTs are required.

IV. RESULT

Cost analysis is calculated for each module in the project. Figure 2 shows the overall structure of this. Total cost analysis can be largely calculated as the sum of three costs. The three costs are road maintenance costs, vehicle fuel costs and labor costs to run the project.

First, road maintenance costs can be calculated from data from the following analysis. The maximum rotation angle region obtained through the Curvature of the road analysis has a great influence on whether a vehicle carrying a real module can pass, which is a factor in calculating the road maintenance cost. This is because a module large enough to require many vehicles needs to widen the road for large turning angles. In addition, in the case of a slope data area of 6-7 degrees or more obtained from the slope of the road analysis, the slope repair of the road is required to move the module. Therefore, it acts as a factor of road maintenance cost calculation. In addition, depending on the number and type of obstacles present in the path, the actual vehicle must be removed in order to pass, which is another factor of the road maintenance cost. Lastly, if the road width is narrow, widening work is required, so the regional data obtained from road width analysis is also a factor of road maintenance cost.

Secondly, the fuel cost of the vehicle influences the calculation by the number of SPMT vehicles used to move the module obtained from the transporter analysis, the fuel

 ISBN: 978-1-61208-756-6

cost and the total distance of the route the module is transported.

Finally, labor costs can be calculated from the number of project participants and the project duration. However, since it does not yet provide an input for elements of project headcount and project period input, it is not currently used as a cost analysis result.

V. CONCLUSION

The P-TAIS system introduced in this paper is a system used for land transportation of modular plants, providing management of each project and module, and recommending and analyzing the route for transporting modules. This allows project managers to assess the possibility of transportation projects through P-TAIS, and analyze the costs required before contracting the plant project to ensure that the correct plant orders are achieved.

The system is not yet perfect, so there are some things that need to be fixed. Especially in estimating costs, more analysis is needed. This needs to be closer to realistic cost calculation through much communication with the company. There is also a need to expand and update the analyzes to further refine these factors.

REFERENCES

[1] M. Rashed and G. Heravi. “A lean management approach for power plant construction projects: Wastes identification and assessment.” Proceedings for the 20th Annual Conference of the IGLC, San Diego, USA. 2012.

[2] J. M. Marshall and P, Navarrp, “Costs of nuclear power plant construction: theory and new evidence.” The Rand journal of economics, pp. 148-154, 1991.

[3] J. T. O’Connor, W. H. O’Brien and J.O. Choi. “Industrial project execution planning: Modularization versus stick-built.” Practice periodical on structural design and construction, vol. 21-1, 2015.

[4] J. Choi and H. J. Song. “Evaluation of the modular method for industrial plant construction projects.” International Journal of Construction Management, vol. 14-3, pp. 171-180, 2014.

[5] P. J. Noble. “Process plant Construction.” Wiley & Sons, Ltd. Publication. 2009.

[6] Plant-Transportation Analysis Information System, P-TAIS: [online] Available : https://ptais.yonsei.ac.kr. 2019.10.8.

[7] R. Joseph and F. Ali. “YOLOv3: An incremental improvement.” arXiv, 2018.

[8] P. Isola, J. Y. Zhu, T. Zhou and A. A. Efros. “Image-to-image translation with conditional adversarial networks.” Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1125-1134, 2017.

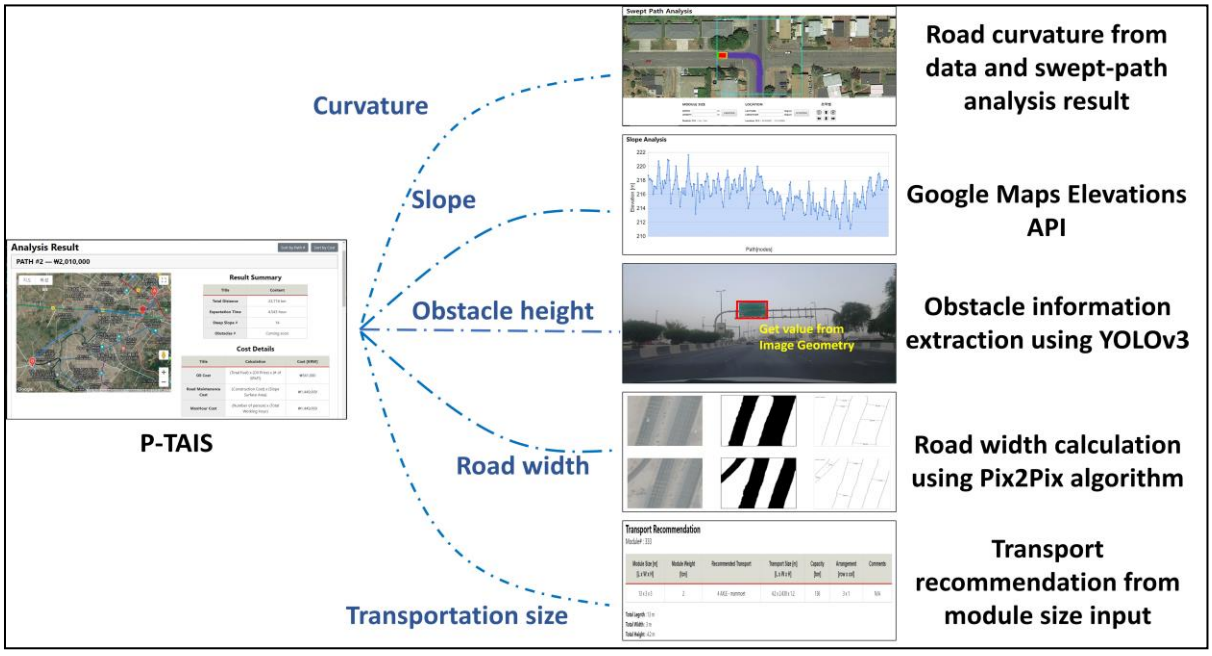


Figure 2. P-TAIS Cost Analysis Structure