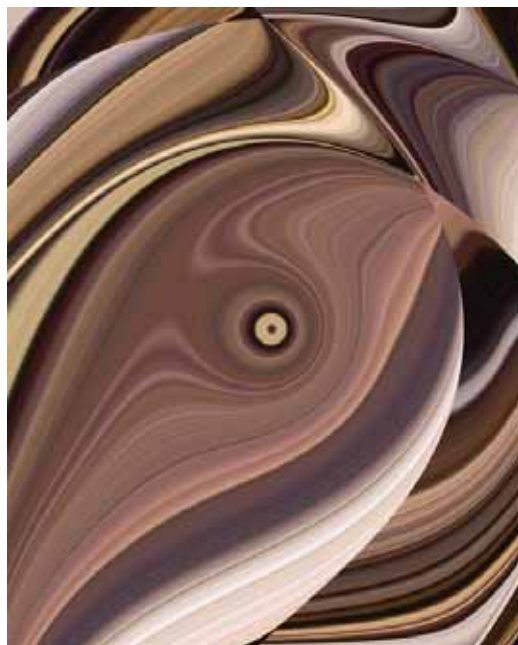


26. ASIM Symposium Simulationstechnik

25.7.-27.7.2022, TU Wien

Herausgegeben von
Felix Breitenacker, Christina
Deatcu, Umut Durak, Andreas
Körner
und Thorsten Pawletta



ASIM SST 2022



Proceedings Kurzbeiträge





ASIM



ASIM



ASIM



ASIM Books – ASIM Book Series – ASIM Buchreihen

Proceedings

- Proceedings Langbeiträge ASIM SST 2022 -26. ASIM Symposium Simulationstechnik**, TU Wien, Juli 2022
F. Breitenecker, C. Deatcu, U. Durak, A. Körner, T. Pawletta (Hrsg.), ARGESIM Report 20; ASIM Mitteilung AM 181
ISBN ebook 978-3-901608-97-1, DOI 10.11128/arep.20, ARGESIM Verlag Wien, 2022; ISBN print 978-3-903311-19-0, TU Verlag
- Proceedings Kurzbeiträge ASIM SST 2022 -26. ASIM Symposium Simulationstechnik**, TU Wien, Juli 2022
F. Breitenecker, C. Deatcu, U. Durak, A. Körner, T. Pawletta (Hrsg.), ARGESIM Report 19; ASIM Mitteilung AM 179
ISBN ebook 978-3-901608-96-4, DOI 10.11128/arep.19, ISBN print 978-3-901608-73-5, ARGESIM Verlag Wien, 2022
- Simulation in Production and Logistics 2021 – 19. ASIM Fachtagung Simulation in Produktion und Logistik**
Online Tagung, Sept. 2021, J. Franke, P. Schuderer (Hrsg.), Cuvillier Verlag, Göttingen, 2021,
ISBN print 978-3-73697-479-1; ISBN ebook 978-3-73696-479-2; ASIM Mitteilung AM177
- Proceedings ASIM SST 2020 – 25. ASIM Symposium Simulationstechnik**, Online-Tagung
14.-15.10.2020; C. Deatcu, D. Lückerrath, O. Ullrich, U. Durak (Hrsg.), ARGESIM Verlag Wien, 2020;
ISBN ebook: 978-3-901608-93-3; DOI 10.11128/arep.59; ARGESIM Report 59; ASIM Mitteilung AM 174
- Simulation in Production and Logistics 2019 – 18. ASIM Fachtagung Simulation in Produktion und Logistik**
Chemnitz, 18.-20. 9. 2019; M. Putz, A. Schlegel (Hrsg.), Verlag Wissenschaftliche Skripten Auerbach, 2019,
ISBN print 978-3-95735-113-5, ISBN ebook 978-3-95735-114-2; ASIM Mitteilung AM172
- Tagungsband ASIM SST 2018 - 24. ASIM Symposium Simulationstechnik**, HCU Hamburg, Oktober 2018
C. Deatcu, T. Schramm, K. Zobel (Hrsg.), ARGESIM Verlag Wien, 2018; ISBN print: 978-3-901608-12-4;
ISBN ebook: 978-3-901608-17-9; DOI 10.11128/arep.56; ARGESIM Report 56; ASIM Mitteilung AM 168

Book Series Fortschrittsberichte Simulation – Advances in Simulation

- Cooperative and Multirate Simulation: Analysis, Classification and New Hierarchical Approaches.** I. Hafner, FBS 39
ISBN ebook 978-3-903347-39-7, DOI 10.11128/fbs.39, ARGESIM Publ. Vienna, 2022; ISBN print 978-3-903311-07-7, TUVerlag Wien, 2022
- Die Bedeutung der Risikoanalyse für den Rechtsschutz bei automatisierten Verwaltungsstrafverfahren.** T. Preiß, FBS 38
ISBN ebook 978-3-903347-38-0, DOI 10.11128/fbs.38, ARGESIM Publ. Vienna, 2020; ISBN print 978-3-903311-14-5, TUVerlag Wien, 2020
- Methods for Hybrid Modeling and Simulation-Based Optimization in Energy-Aware Production Planning.** B. Heinzl, FBS 37
ISBN ebook 978-3-903347-37-3, DOI 10.11128/fbs.37, ARGESIM Publ. Vienna, 2020; ISBN print 978-3-903311-11-4, TUVerlag Wien, 2020
- Konforme Abbildungen zur Simulation von Modellen mit verteilten Parametern.** Martin Holzinger, FBS 36
ISBN ebook 978-3-903347-36-6, DOI 10.11128/fbs.36, ARGESIM Publ. Vienna, 2020; ISBN print 978-3-903311-10-7, TUVerlag Wien, 2020
- Fractional Diffusion by Random Walks on Hierarchical and Fractal Topological Structures.** G. Schneckenreither, FBS 35
ISBN ebook 978-3-903347-35-9, DOI 10.11128/fbs.35, ARGESIM Publ. Vienna, 2020
- A Framework Including Artificial Neural Networks in Modelling Hybrid Dynamical Systems.** Stefanie Winkler, FBS 34
ISBN ebook 978-3-903347-34-2, DOI 10.11128/fbs.34, ARGESIM Publ. Vienna, 2020; ISBN print 978-3-903311-09-1, TUVerlag Wien, 2020
- Modelling Synthesis of Lattice Gas Cellular Automata and Random Walk and Application to Gluing of Bulk Material.** C. Rößler, FBS 33
ISBN ebook 978-3-903347-33-5, DOI 10.11128/fbs.33, ARGESIM Publ. Vienna, 2020; ISBN print 978-3-903311-08-4, TUVerlag Wien, 2020
- Combined Models of Pulse Wave and ECG Analysis for Risk Prediction in End-stage Renal Disease Patients.** S. Hagmair, FBS 32
ISBN ebook 978-3-903347-32-8, DOI 10.11128/fbs.32, ARGESIM Publ. Vienna, 2020
- Mathematical Models for Pulse Wave Analysis Considering Ventriculo-arterial Coupling in Systolic Heart Failure.** S. Parragh, FBS 31
ISBN ebook 978-3-903347-31-1, DOI 10.11128/fbs.31, ARGESIM Publ. Vienna, 2020
- Variantenmanagement in der Modellbildung und Simulation unter Verwendung des SES/MB Frameworks.** A. Schmidt, FBS 30;
ISBN ebook 978-3-903347-30-4, DOI 10.11128/fbs.30, ARGESIM Verlag, Wien 2019; ISBN print 978-3-903311-03-9, TUVerlag Wien, 2019
- Classification of Microscopic Models with Respect to Aggregated System Behaviour.** Martin Bicher, FBS 29;
ISBN ebook 978-3-903347-29-8, DOI 10.11128/fbs.29, ARGESIM Publ. Vienna, 2017; ISBN print 978-3-903311-00-8, TUVerlag Wien, 2019
- Model Based Methods for Early Diagnosis of Cardiovascular Diseases.** Martin Bachler, FBS 28;
ISBN ebook 978-3-903347-28-1, DOI 10.11128/fbs.28, ARGESIM Publ. Vienna, 2017; ISBN print 978-3-903024-99-1, TUVerlag Wien, 2019
- A Mathematical Characterisation of State Events in Hybrid Modelling.** Andreas Körner, FBS 27;
ISBN ebook 978-3-903347-27-4, DOI 10.11128/fbs.27, ARGESIM Publ. Vienna, 2016
- Comparative Modelling and Simulation: A Concept for Modular Modelling and Hybrid Simulation of Complex Systems.** N. Popper, FBS 26;
ISBN ebook 978-3-903347-26-7, DOI 10.11128/fbs.26, ARGESIM Publ. Vienna, 2016
- Rapid Control Prototyping komplexer und flexibler Robotersteuerungen auf Basis des SBE-Ansatzes.** Gunnar Maletzki, FBS 25;
ISBN ebook 978-3-903347-25-0, DOI 10.11128/fbs.25, ARGESIM Publ. Vienna, 2019; ISBN Print 978-3-903311-02-2, TUVerlag Wien, 2019
- A Comparative Analysis of System Dynamics and Agent-Based Modelling for Health Care Reimbursement Systems.** P. Einzinger, FBS 24;
ISBN ebook 978-3-903347-24-3, DOI 10.11128/fbs.24, ARGESIM Publ. Vienna, 2016
- Agentenbasierte Simulation von Personenströmen mit unterschiedlichen Charakteristiken.** Martin Bruckner, FBS 23;
ISBN ebook Online 978-3-903347-23-6, DOI 10.11128/fbs.23, ARGESIM Verlag Wien, 2016
- Deployment of Mathematical Simulation Models for Space Management.** Stefan Emrich, FBS 22;
ISBN ebook 978-3-903347-22-9, DOI 10.11128/fbs.22, ARGESIM Publisher Vienna, 2016
- Lattice Boltzmann Modeling and Simulation of Incompressible Flows in Distensible Tubes for Applications in Hemodynamics.** X. Descovich, FBS 21; ISBN ebook 978-3-903347-21-2, DOI 10.11128/fbs.21, ARGESIM, 2016; ISBN Print 978-3-903024-98-4, TUVerlag 2019
- Mathematical Modeling for New Insights into Epidemics by Herd Immunity and Serotype Shift.** Florian Miksch, FBS 20;
ISBN ebook 978-3-903347-20-5, DOI 10.11128/fbs.20, ARGESIM Publ. Vienna, 2016; ISBN Print 978-3-903024-21-2, TUVerlag Wien, 2016
- Integration of Agent Based Modelling in DEVS for Utilisation Analysis: The MoreSpace Project at TU Vienna.** S. Tauböck, FBS 19
ISBN ebook 978-3-903347-19-9, DOI 10.11128/fbs.19, ARGESIM Publ., 2016; ISBN Print 978-3-903024-85-4, TUVerlag Wien, 2019

ASIM SST 2022

Proceedings Kurzbeiträge

26. ASIM Symposium Simulationstechnik
25.7.-27.7.2022, TU Wien

Herausgegeben von
Felix Breitenecker, Christina Deatcu, Umut Durak,
Andreas Körner und Thorsten Pawletta

ISBN print
ISBN 978-3-901608-73-5
ARGESIM Verlag, Wien, 2022
www.argesim.org

ARGESIM Report 19

ISBN ebook
ISBN 978-3-901608-96-4
10.11128/arep.19
ARGESIM Verlag, Wien, 2022
www.argesim.org

ASIM Mitteilung 179

Bibliographic Data:

Publisher: ARGESIM Publisher, Vienna

Title: Proceedings Kurzbeiträge ASIM SST 2022

Subtitle: 26. ASIM Symposium Simulationstechnik, 25.7.-27.7.2022, TU Wien

Author(s):

Editor(s): Felix Breitenecker, Christina Deatcu, Umut Durak, Andreas Körner, Thorsten Pawletta

Series: ARGESIM Reports

Series Editors: Felix Breitenecker, Thorsten Pawletta, ASIM

Volume: ARGESIM Report no. 19

ISBN print: 978-3-901608-73-5, ARGESIM Verlag

ISBN ebook: 978-3-901608-96-4, ARGESIM Verlag

DOI: 10.11128/arep.19

ASIM ID: ASIM Mitteilung 179

Publication Date: July 20, 2022

Number of Pages: 84 +viii pages

Cover: 2 reproductions of algorithmic art graphic series 'Metamorphosis' by Vlatko Ceric, www.vceric.net, with permission of the artist

Copyright © 2022 ARGESIM Publisher

Copyright Information / Regulations ARGESIM

ARGESIM is a non-profit scientific society generally aiming for dissemination of information on system simulation - from research via development to applications of system simulation. ARGESIM's primary publication is the journal SNE – Simulation Notes Europe with open access to all contributions; generally, the authors retain the copyright of their SNE contributions. This copyright regulation holds also for *ARGESIM Reports* and *ARGESIM Advances in Simulation* publishing conference volumes for ASIM, MATHMOD, and EU-ROSIM (in consideration of copyright regulations for related conference publications) and monographs on system simulation (mainly PhD theses).

About ARGESIM

ARGESIM is a non-profit society generally aiming for dissemination of information on system simulation from research via development to applications of system simulation. ARGESIM is closely co-operating with EU-ROSIM, the Federation of European Simulation Societies, and with ASIM, the German Simulation Society. ARGESIM is an 'outsourced' activity from the Mathematical Modelling and Simulation Group of TU Wien, there is also close co-operation with TU Wien (organisationally and personally). *ARGESIM Publisher* organizes publishing activities, with ISBN root 978-3-901608-xx-y and DOI root 10.11128/xx...x.

ARGESIM's activities are:

- Publication of the scientific journal SNE - Simulation Notes Europe (Membership Journal of EUROSIM, the Federation of European Simulation Societies) → www.sne-journal.org
- Organisation and Publication of the ARGESIM Benchmarks for Modelling Approaches and Simulation Implementations → www.argesim.org/benchmarks/
- Publication of the series ARGESIM Reports (for monographs in system simulation, and proceedings of simulation conferences and workshops) → www.argesim.org/publications/
- Publication of the special series FBS Simulation - Advances in Simulation / Fortschrittsberichte Simulation (monographs in co-operation with ASIM, the German Simulation Society)
- Organisation of the Conference Series MATHMOD Vienna (triennial, in co-operation with EUROSIM, ASIM, and TU Wien) → www.mathmod.at
- Administration and support of ASIM (German Simulation Society → www.asim-gi.org) and of EUROSIM (Federation of European Simulation Societies → www.eurosim.info)

ARGESIM – Arbeitsgemeinschaft Simulation News – Working Committee Simulation News – SNE Publication
Mommengasse 19/8, 1040 Vienna, Austria; Tel +43-1-58801-10111, -10115; Fax +43-1-58801-910111
Email: office@argesim.org, office@sne-journal.org; WWW: www.argesim.org, www.sne-journal.org
Incorporated Austrian Society ZVR No 213056164 – EU VAT ID No ATU 72054279
Bank Account: ARGESIM, IBAN AT07 2011 1828 9115 0800, BIC GIBAATWWXXX, ERSTE BANK VIENNA

VORWORT

ASIM lädt zu einer Tagung ein, wie es früher einmal war – lautet das Motto der Veranstalter von **ASIM SST 2022**, dem **26. ASIM Symposium Simulationstechnik**. Nach einigen virtuellen und hybriden Tagungen war es der erklärte Wunsch, wieder einmal eine ASIM-Präsenztagung zu veranstalten.

Traditionell findet das **ASIM Symposium Simulationstechnik** alle zwei Jahre im Herbst statt. Das Jubiläumssymposium, das **25. ASIM Symposium Simulationstechnik** konnte im Herbst 2020 nur virtuell veranstaltet werden. Planungen für das 26. Symposium sahen einerseits den Herbst 2022 vor, andererseits entstand die Idee, den Frühjahrs-Workshop der ASIM-Fachgruppen GMMS und STS gemeinsam mit dem Symposium Simulationstechnik zu veranstalten, und darüber hinaus auch noch mit der Tagung **MATHMOD Vienna** zu verbinden, deren Co-Sponsor ASIM ist. Die **10th MATHMOD Vienna** musste Corona-bedingt bereits von Februar 2021 auf Februar 2022 verschoben werden – und daher entschieden sich die Organisatoren für **ASIM SST 2022** im Februar 2022, direkt vor der **MATHMOD 2022**.

Wieder machte Corona einen Strich durch die Planung, ein weiteres Verschieben auf den Herbst 2022 erwies sich als ungünstig, da der Herbst 2022 mit Tagungen vollgepfastert ist (sehr viele Corona-bedingte Verschiebungen) – eine gemeinsame Planung mit den **MATHMOD** Organisatoren (teilweise in Personalunion mit den **ASIM SST 2022** Organisatoren) führte daher zu **ASIM SST 2022** und **MATHMOD Vienna 2022** im Juli 2022 - **Tagungen, wie es früher einmal war - conference as it used to be**.

ASIM SST 2022 bietet zwei Arten von Beiträgen, die alle einem Peer Review Prozess unterzogen wurden und sich in der Publikationsform unterscheiden:

- **ASIM SST 2022 Langbeiträge** (6-10 Seiten, deutsch oder englisch) mit individueller DOI für jeden Beitrag im Print- und e-Proceedings Band:
Proceedings Langbeiträge ASIM SST 2022, ARGESIM Report 20, ASIM Mitteilung 180, ISBN ebook 978-3-901608-97-1 ARGESIM Verlag,
ISBN print 978-3-903311-19-0, TU Verlag, Print on Demand
- **ASIM SST 2022 Kurzbeiträge** (2-5 Seiten, deutsch oder englisch), mit Tagungsband-DOI:
Proceedings Kurzbeiträge ASIM SST 2022, ARGESIM Report 19, ASIM Mitteilung 179, ISBN ebook 978-3-901608-96-4 ARGESIM Verlag,
ISBN print 978-3-901608-73-5, ARGESIM Verlag

Wie bei allen ASIM SST und MATHMOD Tagungen bietet **ASIM SST 2022** auch (gemeinsame) Tutorials an. Zudem haben die Organisatoren eine Einrichtung früherer ASIM-Tagungen wiedereingeführt, das **Forum Simulationstechnik**: kontroverse Impulsvorträge zu Entwicklungen in der Simulationstechnik von Simulationsexperten (aus ASIM und Gäste; s. Seite iv) laden zu einer regen Diskussion ein, verbunden mit einer „Strudelsitzung“. Und natürlich freut es uns, dass wir zwei ASIM-Mitglieder für Hauptvorträge gewinnen konnten, Dirk Zimmer (DLR Oberpfaffenhofen; Modelica Mitentwickler), und Niki Popper (TU Wien & dwh Simulation Services; Mitglied des Corona-Beratungsgremiums der österreichischen Bundesregierung).

Zum **ASIM SST 2022** haben sich etwa 80 Fachleute zusammengefunden, zwei Drittel davon ASIM-Mitglieder. Dieser Tagungsband, die *Proceedings Kurzbeiträge ASIM SST 2022* publiziert die 16 angenommenen Kurzbeiträge – mit Volume DOI, gruppiert nach thematischen Sessions. Wie bei jeder ASIM-Tagung, ist die Möglichkeit für Postconference Publications in EUROSIM's bzw. ASIM's Scientific Journal *Simulation Notes Europe SNE* vorgesehen.

ASIM lässt bei Tagungen und Workshops den Autorinnen und Autoren bewusst die Wahl der Sprache frei: Englisch für weitere Publikationsverbreitung, oder Deutsch für weitere Kommunikation innerhalb der ASIM-Fachgruppen. Daher sind in den Proceedings beide Sprachen bunt gemischt vertreten.

Virtuelle und hybride Tagungen haben zu einer Verarmung der gesellschaftlichen Aspekte von Tagungen geführt. Der Wunsch nach einer Präsenztagung für **ASIM SST 2022** entstand daher auch aus dem Bedürfnis, diese „Verarmung“ zu beenden, und bei Tagungen wieder wissenschaftlich-gesellschaftliche Kontakte pflegen zu können. Daher bietet **ASIM SST 2022** auch wieder Gesellschaftliches: *Abendveranstaltungen*, das *Cafe Simulation*, *Kunst*, und traditionellerweise *besondere Tagungstaschen* und den *ASIM Tagungswein*, sowie das *Forum Simulationstechnik mit Begleitung*:

- **Tagungsabend Café Resselpark:** ASIM lädt dazu in ein geschichtsträchtiges Lokal am Karlsplatz ein (Teilnehmer können Harry Lime / Orson Welles treffen), zu Wiener Schmankeleien ein, und zur Verkostung der Tagungsweine.
- **Farewell Party ASIM – Welcome Party MATHMOD:** **ASIM SST 2022** Teilnehmer und **MATHMOD 2022** Teilnehmer haben die Gelegenheit, im *Foyer Gusshaus* der TU Wien bei Snacks und Getränken Erfahrungen auszutauschen
- **Cafe Simulation – Ausstellung Algorithmic Art:** Nahe dem Tagungsbüro bietet das *Cafe Simulation* ein Kommunikationszentrum an, mit typischem Wiener Kaffee von Mokka bis Melange (neben den bewusst langen Kaffeepausen), mit Kostproben des Tagungsweins, und mit Algorithmic Art – Grafiken von Vlatko Čerić (die auch auf den **ASIM SST 2022** Publikationen, auf den Tagungsbadges, etc. zu finden sind).
- **Tagungstaschen:** ASIM- und MATHMOD-Tagungen in Wien pflegen eine besondere Tradition für Tagungstaschen/Tagungsboxen, sie werden von Behindertenwerkstätten angefertigt. Die Rucksack-ähnlichen Sportbeutel als Tagungstaschen für **ASIM SST 2022** und **MATHMOD 2022** haben betreute Personen der Behindertenwerkstatt Ternitz mit den Tagungsnamen bestickt.
- **Tagungswein:** Bereits bei frühen ASIM-Tagungen in Wien (z.B. 1984), beim EUROSIM Congress 1995 in Wien, und bei vielen MATHMOD-Tagungen hat der Tagungswein Gefallen gefunden – bei Kostproben und als Mitgabe für jeden Tagungsteilnehmer. An diese Tradition schließt **ASIM SST 2022** an, mit Wein vom Weingut Zadrazil aus Retz (NÖ), von Weinsorten *Grüner Veltliner* bis *Gemischter Satz*, und mit der Weinsorten-Besonderheit *Gemischte Katz*.
- **Strudelsitzung:** Warum nicht Wissenschaft mit Kaffee und Kuchen verbinden – *Das Forum Simulationstechnik*, der letzte Programmpunkt von **ASIM SST 2022**, hat als Begleitung eine „Strudelsitzung“: Diskutanten können sich an ‚Ausgezogenen Strudeln‘ und Kaffee und Getränken stärken.

Wir Organisatoren möchten allen herzlich danken, die das **ASIM SST 2022** zu einem Erfolg machen - *zu einer Tagung ein, wie es früher einmal war*: den Teilnehmerinnen und Teilnehmern, den Autoren der Beiträge, dem Programmkomitee für das Reviewing, dem ARGESIM Verlag für die Publikationen, und unseren Tagungsmitarbeitern.

*Felix Breitenacker, Christina Deatcu, Umut Durak,
Andreas Körner und Thorsten Pawletta, Juli 2022*

ASIM SST 2022

26. ASIM Symposium Simulationstechnik, 25.7.-27.7.2022, TU Wien

Sponsor und Organisation - Sponsored and Organized by

TU Wien, Inst. f. Analysis and Scientific Computing

Co-Sponsoren - Co-Sponsors

- **ASIM** – German Simulation Society (Arbeitsgemeinschaft Simulation) → www.asim-gi.org
- **ARGESIM** Vienna – Arbeitsgemeinschaft Simulation News, Wien → www.argesim.org
- **EUROSIM** – Federation of European Simulation Societies → www.eurosim.info

Programmkomitee – Programme Committee

Umut Durak, DLR (Chair)

Felix Breitenecker, TU Wien, (Co-Chair)

Christina Deatcu, HS Wismar, (Co-Chair)

Kurt Chudej, Universität Bayreuth

Walter Commerell, TH Ulm

Umut Durak, DLR

Leo Gall, LTX Simulation GmbH, München

Lukas Hollenstein, HS Zürich

Andreas Körner, TU Wien

Christoph Laroque, Hochschule Zwickau

Xiabo Liu-Henke, Hochschule Ostfalia

Daniel Lückerrath, Fraunhofer IAIS

Heinz-Theo Mammen, Hella KGaA
Hueck & Co, Lippstadt

Thorsten Pawletta, Hochschule Wismar

Nicolas Popper, dwh GmbH, Wien

Oliver Rose, Univ. der Bundeswehr München

Thomas Schramm, HCU Hamburg

Michael Striebel, HTWG Konstanz

Oliver Ullrich, Fraunhofer IAIS

Sigrid Wenzel, Universität Kassel

Jochen Wittmann, HTW Berlin

Organisationskomitee – Organizing Committee

Andreas Körner (Chair)

Felix Breitenecker (Co-Chair)

Brigitte Ecker (Conf. Ass.)

ASIM 2022 Hauptvorträge - ASIM 2022 Invited Lectures

- Robuste Modellierung und informationsverlustbehaftete Idealisierung in objekt-orientierter Modellierung physikalischer Systeme
Dirk Zimmer, DLR
- Agent-based COVID-19 Simulation: General Analysis and Guideline for Politicians !?
Niki Popper, TU Wien / DWH Simulation Services

ASIM 2022 Forum Simulationstechnik - ASIM 2022 Forum Simulation Technique

Die ASIM 2022, das 26. ASIM Symposium Simulationstechnik, setzt mit dem **Forum Simulationstechnik** eine Tradition früherer ASIM-Tagungen fort. Das Forum bringt (kontroverse) Impulsvorträge zu Entwicklungen in der Simulationstechnik von Simulationsexperten (aus ASIM und Gäste); Teilnehmer sind zu eigenen Impulsvorträgen und/oder zur Diskussion der Impulsvorträge eingeladen:

- Lieber KI mit vielen simulierten Daten als KI mit wenig realen Daten !
O. Rose
- System Dynamics - Modellbildung nur für Dummies ?
P. Junglas
- Co-Simulation - Methode zur Unterstützung der Faulheit in der Modellierung ?
F. Breitenecker, I. Hafner, N. Popper
- Reinforcement Learning - Wettstreit Model-based vs Model-free ?
D. Pasterk, M. Wastian
- Daten für Corona-Simulationen - von exzellent über heimtückisch bis unbrauchbar
N. Popper
- ARGESIM Benchmarks - Out of Date oder Tool für die Simulationslehre ?
F. Breitenecker, P. Junglas, C. Deatcu
- und weitere

ASIM 2022 Thematische Sessions - ASIM 2022 Thematic Sessions

- Simulation in Produktion und Logistik - Simulation in Production and Logistics
- Diskrete Systeme - Discrete Systems
- Simulation Technischer Systeme - Simulation of Technical Systems
- Simulation Biologie, Medizin und Umwelt - Simulation in Biology, Medicine, and Environment
- Simulationsmethoden - Methods in Simulation

Inhaltsverzeichnis – List of Content

HAUPTVORTRÄGE - INVITED LECTURES	1
Agent-based COVID-19 Simulation: General Analysis and Guideline for Politicians !? <i>Niki Popper</i>	3
Robuste Modellierung und informationsverlustbehaftete Idealisierung in objekt-orientierter Modellierung physikalischer Systeme <i>Dirk Zimmer</i>	5
SIMULATION IN PRODUKTION UND LOGISTIK – SIMULATION IN PRODUCTION AND LOGISTICS	7
Operative, simulationsgestützte Leistungssteuerung von Springern in einer getakteten Endmontagelinie <i>L. März</i>	9
Simulationsgestütztes Line Balancing <i>L. März</i>	13
Simulating the Viability of Crowd Logistics in Short Food Supply Chains <i>F. Cramer, C. Fikar</i>	17
Kopplung einer Material- und Energieflusssimulation mit Reinforcement-Learning-Algorithmen <i>J. Dettelbacher, D. Wagner, A. Buchele, W. Schlüter</i>	21
Erstellung eines Verkehrssimulationsmodells für Berlin in SUMO zur Simulation der möglichen Emissionen und daraus resultierendem Routing <i>M. Engelmann, J. Wittmann, V. Wohlgemuth</i>	25
DISKRETE SYSTEME – DISCRETE SYSTEMS	31
Analysis of Train Delays using Bayesian Networks <i>V. Barzal, M. Rößler, M. Wastian, F. Breitenecker, N. Popper</i>	33
Parameter-Free Approximation Method for Controlling Discrete Event Simulation by Reinforcement Learning <i>D. Pasterk, A. Körner</i>	37
SIMULATION TECHNISCHER SYSTEME – SIMULATION OF TECHNICAL SYSTEM	41
Numerische Simulation einer asymmetrischen Drallströmung - Vergleich der numerischen Turbulenzmodellierung mit experimentellen Messungen <i>K. Zacharias, D. Welsch, W. Schlüter, A. Buchele</i>	43
Modelica Simulation of Pendulums and Crane Crab: PyMbs vs. OpenModelica <i>M. Rößler, M. Ehrmann, F. Breitenecker</i>	47

SIMULATION IN BIOLOGIE, MEDIZIN UND UMWELT	
– SIMULATION IN BIOLOGY, MEDICINE, AND ENVIRONMENT	51
Wirkstoffidentifikation anhand mikrobieller Proteinmuster	53
<i>O. Al-Latif, H. Hillenhagen, J. Bernhardt, A. Grüning, C. Wahmkow</i>	
Die Flugrouten der Graukraniche – eine GIS-basierte Analyse zum Zusammenhang zwischen Routenwahl und Wetterbedingungen	57
<i>S. Schreiber, T. Christiansen, J. Wittmann</i>	
Modelling Blood Volume Oscillations in the Finger	63
<i>H. Kastinger, M. Bachler</i>	
SIR-type Epidemic Models for Education at Web	67
<i>A. Freißlinger, A. Körner, F. Breiteneker</i>	
SIMULATIONSMETHODEN - METHODS IN SIMULATION	71
Geometrisches Mehrgitterverfahren für die gyrokinetische Poisson-Gleichung mit Anwendung auf Kernfusion	73
<i>C. Schwarz, P. Leleux, M. J. Kühn, U. Rude</i>	
Hestia jl: A Julia library for heat conduction modeling with boundary actuation	77
<i>S. Scholz, L. Berger</i>	
Untersuchung der anwendungsspezifischen Verifikations- und Validierungstechniken unter Berücksichtigung einer veränderten Datenbasis	81
<i>K. Langenbach, A. A. Scheidler, M. Rabe</i>	

Autorenindex – Index of Authors

- | | | |
|---------------------------------|--------------------------------|------------------------------|
| N. Adamchuk (L: 235) | B. H. Herrmann (L: 139) | M. Ringel (L: 139) |
| O. Al-Latif 53 | H. Hillenhagen 53 | B. Rösch (L: 235) |
| V. Ahlers (L: 263) | S. Jacobitz (L: 147, 253) | M. Rößler 33, 47 |
| M. Bachler 63 | D. Jammer (L: 93) | U. Rüde 73 |
| V. Barzal 33 | U. Jessen (L: 39) | A. A. Scheidler 81 |
| J. Baumgartner (L: 235) | D. Jörss (L: 139) | W. Schlüter 21, 43, (L: 235) |
| L. Berger 77 | P. Junglas (L: 57, 93) | L. Schmedes (L: 57) |
| S. Bergmann (L: 15, 65) | M. Kalenberg (L: 171) | A. Schmid (L: 73) |
| J. Bernhardt 53 | F. Kamhuber (L: 73) | G. Schneider (L: 49) |
| L. Bienzeisler (L: 263) | H. Kastinger 63 | W. Scholl (L: 49) |
| D. Bodnár (L: 101) | A. Körner 37, 67 | S. Scholz 77 |
| J. Bolz (L: 197) | A. Koschel (L: 263) | S. Schreiber 57 |
| F. Breitenacker 33, 47, 67 | P. Krenckel (L: 211) | C. Schwarz 73 |
| A. Buchele 21, 43 | C. Krull (L: 101, 109, 211) | M. Shakoorianfard (L: 155) |
| R. Büchi (L: 179, 205) | M. J. Kühn 73 | W. Sihn (L: 73) |
| T. Christiansen 57 | K. Langenbach 81 | T. Sobottka (L: 73) |
| S.P. Christiansen-Lenger (L: 9) | C. Laroque (L: 49) | S. Spieckermann (L: 39) |
| S. Cortès (L: 119, 129) | M. Leißau (L: 49) | N. Stahl (L: 163) |
| F. Cramer 17 | T. Lelke (L: 263) | S. Straßburger (L: 65, 243) |
| J. Dettelbacher 21 | P. Leleux 73 | R. Sutherland (L: 39) |
| C. Dettmann (L: 119, 129) | A. Lepp (L: 187) | O. Ullrich (L: 81) |
| U. Durak (L: 219) | X. Liu-Henke (L: 31, 147, 253) | T. Vietor (L: 31) |
| M. Ehrmann 47 | D. Lückerrath (L: 81) | D. Wagner 21 |
| M. Engelmann 25 | N. Marheineke (L: 163) | C. Wahmkow 53 |
| S. Emrich (L: 227) | L. März 9, 13 | M. Wastian 33 |
| N. Feldkamp (L: 65) | C. May (L: 171) | V. Waurich (L: 155) |
| C. Fikar 17 | F. Millberg (L: 129) | D. Weidemann (L: 187) |
| C. Fink (L: 139) | K. Nickel (L: 197) | D. Welsch 43 |
| H. Folkerts (L: 219) | F. Özkul (L: 39) | S. Wenzel (L: 39) |
| J. Franke (L: 171) | D. Pasterk 37 | F. Will (L: 155) |
| A. Freißlinger 67 | S. Pawletta (L: 93) | J. Wittmann 25, 57 |
| J. Fritz (L: 147) | T. Pawletta (L: 93, 219) | B. Woerrlein (L: 243) |
| S. Galka (L: 23) | N. Popper 3, 33, (L: 3, 227) | V. Wohlgemuth 25 |
| M. Göllner (L: 253) | P. Preuß (L: 49) | Or Aviv Yarom (L: 147) |
| A. Grüning 53 | R. Pump (L: 263) | K. Zacharias 43 |
| I. Hafner (L: 227) | M. Rabe 81, (L: 9) | J. Zhang (L: 31) |
| O. Haidar (L: 119) | F.-C. Reissner (L: 235) | Dirk Zimmer 5, (L: 5) |
| J. Heib (L: 129) | C. Richter (L: 155) | M. Zwingel 5, (L: 171) |
| L. Heinzelmann (L: 119) | | |

Die Langbeiträge (L) sind zu finden im **ARGESIM Report 20, Proceedings Langbeiträge ASIM SST 2022**

ISBN ebook 978-3-901608-97-1, DOI: 10.11128/arep.20, ASIM Mitteilung AM 180

Hauptvorträge

Invited Lectures

Agent-based COVID-19 Simulation: General Analysis and Guideline for Politicians !?
(Abstract)

Niki Popper, TU Wien / DWH Simulation Services

Robuste Modellierung und informationsverlustbehaftete Idealisierung in
objekt-orientierter Modellierung physikalischer Systeme (Abstract)

Dirk Zimmer, DLR

Agent-based COVID-19 Simulation: General Analysis and Guideline for Politicians !?

Niki Popper

dwh Simulation Services, Neustiftgasse 57-59, 1070 Wien &
TU Wien, Inst.f. Information and Software Engineering, Favoritenstraße 11, 1040 Wien

Abstract. Das Gesundheitssystem ist ein Beispiel für hochkomplizierte dynamische Systeme, die aus Eigendynamik verschiedener Bereiche, den Interessen unterschiedlicher Akteure und Kontrollmechanismen ein kaum nachvollziehbares Amalgam an dynamischen Prozessen und Subsystemen ausbilden. Die COVID-19 Krise und der Umgang damit ist nur ein Beispiel dafür. Modellierung versucht hier ein wenig Licht ins Dunkel zu bringen. Methodische Herausforderungen sind dabei speziell die Kombination komplexer Datenprozesse und unterschiedlicher Modellierungsmethoden. Im Vortrag werden Herausforderungen beleuchtet und Best Practice Beispiele, wie etwa der COVID-19 European Scenario Hub der ECDC präsentiert.

Robuste Modellierung und informationsverlustbehaftete Idealisierung in objekt-orientierter Modellierung physikalischer Systeme

Dirk Zimmer

Deutsches Zentrum für Luft- und Raumfahrt (DLR), Institut für Systemdynamik und Regelungstechnik,
Münchner Straße 20, 82234 Oberpfaffenhofen-Wessling, Deutschland

Abstract. Die objekt-orientierte Modellierung hat zu einer großen Ansammlung von Wissen für cyber-physikalische Systeme geführt. Doch auch nach über 25 Jahren halten sich die Probleme der Skalierung in der täglichen Anwendung hartnäckig. Ohne große Expertise scheitert die Simulation komplexer Systeme immer noch zu häufig. Die Ursachen liegen dabei im Informationsverlust bei der Idealisierung technischer Systeme.

Betrachtet man die Formen der Idealisierung ganz allgemein, fallen gutartige und bösartige Formen auf. Mögen wir lernen, die bösartigen Formen zu erkennen und die gutartigen Formen in methodischer Form zu nutzen. Robuste Modellierungsbibliotheken wie die DLR ThermoFluid Stream Library zeigen die für den Anwender nützlichen Auswirkungen dieser grundlegenden Betrachtungen.

Simulation in Produktion und Logistik

Simulation in Production and Logistics

Optimierung des Designs modularer Montagesysteme (L)

S. Bergmann

Untersuchung von Auftragsfreigabestrategien für mehrzonige Kommissioniersysteme im Online-Lebensmittelhandel (L)

S. Galka

Selbstoptimiertes Auftragsmanagement für autonome Transportfahrzeuge (L)

J. Zhang, X. Liu-Henke, T. Vietor

Operative, simulationsgestützte Leistungssteuerung von Springern in einer getakteten Endmontagelinie (K)

L. März

Simulationsgestütztes Line Balancing (K)

L. März

Verknüpfung von ereignisdiskreter Simulation und Process Mining in Produktion und Logistik (L)

F. Özkul, R. Sutherland, S. Wenzel, U. Jessen, S. Spieckermann

Experimentelle Analyse zur Einschleusplanung in der Halbleiterfertigung mittels einer stochastischen Rückwärtssimulation (L)

C. Laroque, M. Leißau, W. Scholl, G. Schneider, P. Preuß

Discrete event-based modeling of conveyors for dry bulk material (L)

P. Junglas, L. Schmedes

Simulationsbasiertes Deep Reinforcement Learning für Modulare Produktionssysteme (L)

N. Feldkamp, S. Bergmann, S. Straßburger

Simulating the Viability of Crowd Logistics in Short Food Supply Chains (K)

Florian Cramer, Christian Fikar

Kopplung einer Material- und Energieflusssimulation mit Reinforcement-Learning-Algorithmen (K)

J. Dettelbacher, D. Wagner, A. Buchele, W. Schlüter

DISPO 40 - Simulationsgestützte Absatzprognoseoptimierung in der Investitionsgüterindustrie (L)

A. Schmid, F. Kamhuber, T. Sobottka, W. Sihn

Towards Increasing Resilience of Public Transit Infrastructure – A Bi-Modal Simulation Model (L)

D. Lückcrath, Oliver Ullrich

Erstellung eines Verkehrssimulationsmodells für Berlin in SUMO zur Simulation der möglichen Emissionen und daraus resultierendem Routing (K)

M. Engelmann, J. Wittmann, V. Wohlgemuth

Die Langbeiträge (L) sind zu finden im **ARGESIM Report 20**,

Proceedings Langbeiträge ASIM SST 2022

ISBN ebook 978-3-901608-97-1, DOI: 10.11128/arep.20, ASIM Mitteilung AM 180

Operative, simulationsgestützte Leistungssteuerung von Springern in einer getakteten Endmontagelinie

Lothar März^{1*}

¹Stremmer AG, Am Schönbühl 1, 88181 Lindau (Bodensee), Germany; *lothar.maerz@stremmer.de

Abstract. Die Montagelinie von Anhängern für temperierte Waren (Kühlboxen) ist gekennzeichnet von hoher Produktvarianz und flexiblem Mitarbeiterereinsatz. Die Herausforderung der operativen Disposition ist es, die Springer unter Berücksichtigung von Mitarbeiterverfügbarkeiten und Qualifikationen den Optionsprozessen zuzuordnen. Um den Disponenten bei der Planung zu unterstützen, wurde eine simulationsgestützte Springereinsatzplanung realisiert, die neben den Qualifikationen auch Prozessabhängigkeiten und Zeitverluste an den Stationen berücksichtigt. Darüber hinaus werden kontinuierlich der Taktzeitstatus rückgemeldet und bei Änderungen teilweise neue Einsatzpläne generiert, die an die Springer übermittelt werden. Der Beitrag zeigt die Herausforderungen in der Einführung und die Ergebnisse und Nutzen im Einsatz der betrieblichen Praxis auf.

Einführung

Mit einer Jahresproduktion von über 60.000 Fahrzeugen und etwa 6.400 Mitarbeitern ist das Unternehmen Europas führender Hersteller von Sattelauflegern und Anhängern für temperierte Fracht, General Cargo sowie Schüttgüter. Der Großteil der Produktion erfolgt in Deutschland und wird ins Ausland exportiert.

Bei dem Kunden werden Produkte mit hoher Kundenindividualität und komplexer Produktstruktur hergestellt. Die Planung der Produktion hat zur Aufgabe, die hohe Produktvarianz der zu montierenden Auflieger im Tagesverlauf so zu verteilen, dass die eingesetzten Mitarbeiter gleichmäßig ausgelastet und Unterauslastungen sowie Kapazitätsspitzen vermieden werden. Als Produktionssystem für die variantenreiche Serienfertigung haben sich sequenzierte Linien durchgesetzt, bei denen die Produkte taktgebunden eingesteuert werden. Durch die arbeitsintensive Produktionsstruktur und die niedrige Automatisierung ist die Montage ein Bereich mit hohem

Personalaufwand. Neben dem Einkauf ist die Montage daher der Bereich mit den höchsten Einsparpotenzialen.

Die Montage der Kühlboxen erfolgt in einer getakteten Produktion mit teilweise parallelen Arbeitsstationen. In dieser Montage werden ca. 90 Produkte pro Tag mit einer Taktzeit zwischen 13 und 18 Minuten in bis zu drei Schichten gefertigt. Insgesamt verfügt die sequenzierte Montagelinie über 27 Arbeitsstationen in 16 Taktzyklen.

Die vorgestellte Anwendung der Montage Kühlboxen verfolgt das Ziel, die Mitarbeiter im Vorfeld der Montage unter Berücksichtigung der Restriktionen und Randbedingungen so einzuplanen, dass Taktzeitüberschreitungen vermieden werden. Diese simulationsgestützte Funktion zur Feinplanung wird im operativen Ablauf um Rückmeldungen der tatsächlichen Taktzeiten erweitert auf eine Leistungssteuerung.

Der Beitrag gliedert sich in folgende Kapitel:

- Einordnung der Feinplanung und -steuerung in die Planungsarchitektur
- Beschreibung der Restriktionen, Randbedingungen und Wirkzusammenhänge der Montage
- Simulationsbasierte Springereinsatzplanung
- Rückmeldedaten und automatisierte Neuplanung zur Leistungssteuerung der Springer
- Fazit und Ausblick

1 Planungsarchitektur

Die Springereinsatzplanung setzt auf der Sequenzplanung auf. Die Sequenzplanung wird von der Produktionsplanung erstellt und kann bzw. muss aufgrund kurzfristigen Anpassungsbedarfs durch die Montagesteuerung verändert werden. In nachfolgender Abbildung sind die Zusammenhänge zwischen den Planungsebenen dargestellt.

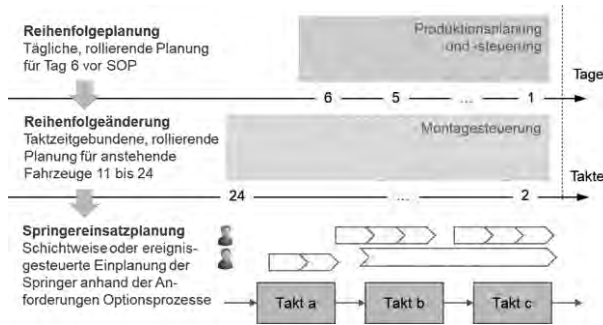


Abbildung 1: Planungshierarchie zur Springereinsatzplanung und -steuerung

In der Endmontage Kühlboxen wird zwischen Standard- und Optionsprozessen unterschieden. Bei den Standardprozessen handelt es sich um Prozesse, welche unabhängig der Produktkonfiguration durch den Kunden bei jedem Trailer durchgeführt werden müssen. Die Standardprozesse werden in der Regel durch sogenannte Standardmitarbeiter durchgeführt, welche jedem Team zugeordnet sind. Die Springer bzw. Optionsmitarbeiter sind nur für optionale Kundenkonfigurationen verantwortlich. Zusätzlich unterstützen sie je nach Bedarf und Möglichkeit Standardmitarbeiter bei den Standardprozessen oder führen zusätzliche Tätigkeiten aus, die regelmäßig anfallen, aber nicht direkt einem Produkt zugeordnet werden können wie z. B. das Auffüllen von Behältern oder Reinigen von Arbeitsmitteln.

Die Einführung der Dynamischen Springerplanung ersetzt die bis dahin verwendeten manuellen Optionstafeln. Auf diesen magnetischen Optionstafeln wurden die nächsten 2-3 Stunden eingeplant. Dieser Planungsansatz fand mit wachsender Komplexität durch die Zunahme wechselseitiger Abhängigkeit der Prozesse seine Grenzen. Darüber hinaus waren damit folgende Einschränkungen verbunden:

- Aufwendige und unpräzise, analoge Planung
- Digital nicht abgreifbare Planung
- Planung nur im Teamleiterbereich möglich
- Keine Berücksichtigung der Reihenfolgestabilität
- Ungenügende Abbildung der Realität (Anzahl Magne)

Die Simulation hat sich zur Unterstützung der Planung von Montagelinien in den letzten Jahren als Stand der Technik etabliert [1]. Zahlreiche Praxisbeispiele in der jüngsten Zeit aus der industriellen Praxis belegen diese These [2, 3] Insbesondere in der Automobilindustrie findet die Methodik zunehmende Einsatzfelder [4] in den unterschiedlichsten Planungsphasen als Assistenz-

systeme [5]. Die Simulation dient dazu, verlässliche Vorhersagen des dynamischen Verhaltens der Mitarbeiter unter Berücksichtigung von beispielsweise Prozessabhängigkeiten zu ermitteln. Simulation ist die einzige Methode, die das Verhalten eines Systems mit zeitlichen Abhängigkeiten bei gleichzeitiger Berücksichtigung einer hohen Anzahl an Planungsparameter in kurzer Zeit vorherzusagen vermag [6]. Im Falle der Springerplanung und -steuerung dient die Simulation zur Vorhersage der Einsätze der Springer an den Stationen. Somit erhält der Disponent eine klare Übersicht über den Arbeitsablauf mit präziser Angabe, welcher Prozess durch welchen Mitarbeiter wann zu bearbeiten ist.

Mit der Einführung der Dynamischen Springerplanung wurden im direkten Vergleich zum vorhergehenden Planungsansatz folgende Ziele verfolgt:

- Berücksichtigung der dynamischen Wirkzusammenhänge durch Simulation und präzise Vorhersage des Personaleinsatzes
- Dialogfähige Planung
- Mit der Systemwelt von SAP integriert
- Teamleiter-übergreifende Einplanung und -steuerung von Springer
- Hohe Reaktionsfähigkeit bei Änderungen im laufenden Betrieb

2 Einflussgrößen der Planung und Steuerung

Aufgrund einer Vielzahl an Einflüssen, Randbedingungen und Restriktionen ist die Vorhersage schwierig, ob mit den verfügbaren Mitarbeitern das anliegende Produktionsprogramm in der vorgesehenen Schicht bzw. innerhalb der Taktzeiten umsetzbar ist. Um präzise planen zu können, sind die Einflussgrößen hinsichtlich ihrer Wirkzusammenhänge im zeitlichen Ablauf zu berücksichtigen. Ausgangspunkt der Planung ist die Produktkonfiguration, aus der sich die Optionsprozesse in Abhängigkeit der Ausstattungsvarianten ergeben. Bei dem Abgleich mit den verfügbaren Kapazitäten wurden nachfolgende Einflussgrößen berücksichtigt.

2.1 Mitarbeiterantritt

Die Mitarbeiter werden im Vorfeld der Montage über die Schichtplanung der Früh-, Spät- oder Nachtschicht zugeteilt. Anhand dieser Personalzuordnung wird die Auslastung und die Machbarkeit der anliegenden Auftragssequenzen analysiert und bewertet.

Es kommt allerdings immer wieder vor, dass Mitarbeiter nicht zur Arbeit erscheinen. Dies ist insbesondere im Zusammenhang mit der Coronapandemie und dem kurzfristigen Ausfall von Mitarbeiter aufgrund positiver Testergebnisse ein häufig auftretendes Ereignis. Um auf diese, erst kurz vor Schichtbeginn offensichtlichen Veränderung der Planungsbasis reagieren zu können, kann der Personalantritt im System/ in der Anwendung entsprechend angepasst und eine Bewertung der Auswirkungen in Echtzeit erfolgen. In Echtzeit beschreibt in diesem Zusammenhang eine Reaktionszeit zur Berechnung und Auswertung der (geänderten) Auslastungssituation von unter 3 Sekunden. Diese kurze Laufzeit ist unbedingte Voraussetzung dafür, um mögliche Alternativszenarien (Sequenzänderungen, Mitarbeiteränderungen o.a.) noch vor Schichtbeginn analysieren und vergleichen zu können.

Ebenfalls spielt die kurze Simulationslaufzeit für die spätere Steuerung während der Montage eine wichtige Rolle, um auf Rückmeldeinformationen unmittelbar zur Steuerung der Springer reagieren zu können.

2.2 Mitarbeiterqualifikation

Nicht jeder Mitarbeiter kann jeden Prozess ausführen. Daher wird in der Applikation eine Liste der Prozess-Stationen-Beziehungen hinterlegt, die aufzeigt, welcher Mitarbeiter für welchen Prozess an welcher Station qualifiziert ist.

Im Zuge der Einführung der Springereinsatzplanung wurde deutlich, dass über die Schnittstelle SAP sukzessive neue Optionsprozesse übermittelt wurden, für die noch keine Mitarbeiter als qualifiziert hinterlegt waren. Um diese Prozesse zu identifizieren, erhält der Planer eine Liste dieser fehlenden Mitarbeiterqualifikationen und kann durch Nachmodellierung die Planung aktuell halten.

Darüber hinaus hat sich die Information über fehlende Qualifikation im Zusammenhang mit dem Mitarbeiterantritt als sehr nützlich erwiesen, denn es kommt immer wieder vor, dass transparent aufgezeigt werden konnte, dass aufgrund kurzfristiger Personalausfälle kein Mitarbeiter mehr vorhanden war, der bestimmte Prozesse hätte ausüben können.

2.3 Prozessabhängigkeiten

Aufgrund der komplexen Zusammenhänge zwischen den Montageprozessen, bei denen ein Prozess erst ausgeführt werden kann, wenn ein oder mehrere andere Prozesse

abgeschlossen sind, ist es notwendig, diese Informationen über Abhängigkeiten mit zu berücksichtigen, da sie einen wesentlichen Einfluss auf die Zeitpunkte der Bearbeitung und damit auf die Verfügbarkeit der Mitarbeiter haben.

Da das Wissen über die Vorgänger-Nachfolger-Beziehungen nicht im Arbeitsplan hinterlegt werden kann (und auch in der Arbeitsplanung nicht vorhanden ist), werden die Prozessbeziehungen in der Planungsanwendung modelliert. Dies erfolgt durch die am Shopfloor agierenden Steuerer, die im direkten Kontakt mit den Workern in der Montage stehen. Sie erhalten kontinuierlich Rückmeldung über die Planungsvorgaben und reichen sukzessive die Planungsgrundlagen durch direkte Eingabe der Prozessabhängigkeiten an.

2.4 Zeitverluste an den Stationen

Um die Kühlboxen von einer Station zu nächsten zu transportieren, werden diese alle zeitgleich in der Linie mittels Fördermittel versetzt. Um die Gesundheit der Mitarbeiter nicht zu gefährden, sind in der Transferzeit keine Tätigkeiten an den Kühlboxen erlaubt. Diese Sperrzeiten müssen bei der Einplanung der Prozesse ebenfalls berücksichtigt werden.

3 Simulationsbasierte Springerplanung

Die Einlastung der Aufträge erfolgt durch die Übernahme der letztgültigen Auftragssequenz. Die Auftragsreihenfolge legt fest, wann welcher Auftrag an welcher Station zu bearbeiten ist. In den Arbeitsplänen sind die an einer Station durchzuführenden Prozesse hinterlegt. Mit der Information, wann welche Sequenzposition an welcher Station anliegt, kann die Auftragslast zyklusgenau vorhergesagt werden. Dies gilt für sequenzierte Produktionslinien, da für diese folgende Voraussetzungen erfüllt sind:

- Getaktete Linie mit einheitlichen Zyklen, d. h. alle Produkte wandern im gleichen Tempo über die Linie
- Die Produktreihenfolge wird über die Sequenz festgelegt. In der Linie kommt es nicht zu Reihenfolgeänderungen, d. h. ein Produkt kann ein anderes nicht überholen, die Reihenfolge bleibt vom Anfang bis Ende stabil.
- Alle Produkte durchlaufen die gleichen Stationen.

In nachfolgender Abbildung sind die notwendigen Informationen beschrieben, die für eine Auftragslastver-

teilung über Stationen und Zyklen notwendig sind.

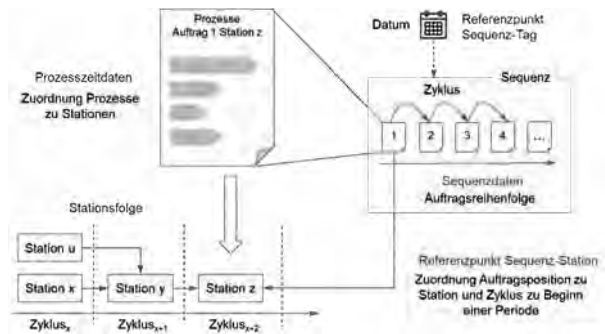


Abbildung 2: Personaleinsatzplanung in sequenzierten Produktionslinien

4 Leistungssteuerung und -optimierung

Die simulationsbasierte Planung ermöglicht eine sehr präzise Vorschau auf das Ablaufverhalten der Mitarbeiter in der Linie. Die Qualität der Planungsergebnisse ist abhängig von der Qualität der Daten. Durch die Verlagerung der Daten- und Prozessmodellierung in die operative Planung wird die Datenqualität sukzessive an die Realität durch die Personen angepasst, die im direkten Kontakt mit den wertschöpfenden Mitarbeitern stehen. Die Letztgenannten geben unmittelbar Rückmeldung über die Korrektheit oder Anpassungsnotwendigkeit der Prozesse und Prozesszeiten. Da der Werkssteuerer diese Anpassungen direkt im Planungssystem umsetzen und eine erneute, simulationsbasierte Planung durchführen kann, entsteht ein kurzzyklischer, positiver Regelkreis zur Erhöhung von Datenqualität und Leistungsoptimierung (Abbildung 3).



Abbildung 3: Beherrschung der Komplexität über zeitnahe Optimierung der Wertschöpfung

Dennoch kann es zu Taktzeitverschiebungen aufgrund ungeplanter Störzeiten kommen. In diesem Falle wird die

tatsächliche Taktzeit an die Simulationsanwendung rückgemeldet und eine neue Planung angestoßen. Die Taktzeitabweichung kann eine Auswirkung auf die Einsatzplanung der Werker in der Linie haben, da verlängerte Taktzeiten beispielsweise die Abarbeitung von Prozessen innerhalb der Taktzeit ermöglichen, die planmäßig in den nächsten Takt per Drift hätten erledigt werden sollen. Zur Steuerung der Mitarbeiter in der Linie sind beispielsweise Tablets oder Smart Watches sinnvolle Devices (siehe Beispiel im Einsatz in [7]), um die kurzfristige Informationsbereitstellung sicherzustellen.

5 Fazit

Der Beitrag zeigt auf, wie die Anwendung zur Dynamischen Springerplanung und -steuerung in der Endmontage Kühlboxen in die operative Planung und Steuerung des Springereinsatzes eingeführt wurde. Anhand einer Live-Demo werden Einsatz der Dynamischen Springerplanung in der Personaleinsatzplanung und -steuerung sowie Herausforderungen und Hemmnisse in der Einführung der Anwendung erläutert. Dem schließt sich ein Erfahrungsbericht und Ausblick zu Nutzen und Weiterentwicklung der Anwendung aus Unternehmensicht an.

Referenzen

- [1] [1] Pinedo, M. Planning and scheduling in manufacturing and service. Corr. As of the 3. Print. Ed. New York: NY: Springer. 2007.
- [2] Mayer, G., Pöge, C., Spiekermann, S., Wenzel, S. editors. Ablaufsimulation in der Automobilindustrie. Springer. 2020
- [3] März, L., Mielke, J. Simulation and Optimization of a Sequenced Mixed-Model Line – Case Study of an Assembly of Gantry Car Washes. Simulation Notes Europe SNE 30(4), 2020, 159-164
- [4] Sommer, A., Steiner, A. Planungsunterstützende Simulationenwerkzeuge kommen in der Automobilindustrie zunehmend zum Einsatz. In J. Bayer, Th. Collisi, S. Wenzel, editors. Simulation in der Automobilproduktion. 2003. p.29-35
- [5] Sokoll, K., Clausen, M. Methoden und Werkzeuge der Simulationsassistenz. In G. Mayer, C. Pöge, S. Spiekermann, S. Wenzel, editors. Ablaufsimulation in der Automobilindustrie. 2020. p.365-377
- [6] März, L. Einsatzmöglichkeiten von Simulation und Optimierung in der Planung der Automobilindustrie. In G. Mayer, C. Pöge, S. Spiekermann, S. Wenzel, editors. Ablaufsimulation in der Automobilindustrie. 2020.
- [7] <https://www.youtube.com/watch?v=1IuPvQ8PUe0> (zuletzt aufgerufen am 11.05.2022)

Simulationsgestütztes Line Balancing

Lothar März^{1*}

¹Stremmer AG, Am Schönbühl 1, 88181 Lindau (Bodensee), Germany; *lothar.maerz@stremmer.de

Abstract. Die Austaktung von sequenzierten Produktionslinien zur bestmöglichen Auslastung der eingesetzten Mitarbeiter ist eine typische Anwendung des Line Balancings. Der Beitrag zeigt auf, dass durch die detaillierte Abbildung der dynamischen Wirkzusammenhänge die Einsatzmöglichkeiten eines simulationsgestützten Line Balancings von der Personalantragsplanung bis zur Einführung neuer Produkte reichen kann. Anhand der Stellgrößen je Anwendungsfall und der Einflussgrößen auf die Fließbandabstimmung werden die Vorteile und der Nutzen dieses Ansatzes anhand von Praxisbeispielen aufgezeigt.

Einführung

Das Line Balancing befasst sich mit dem Planungsproblem der Fließbandabstimmung, das sich bei der Konzeption und dem Betrieb von Fließbändern im Rahmen der Fließbandfertigung stellt. Unter Fließbandfertigung wird in diesem Zusammenhang eine Fertigung verstanden, bei der alle Produkte in der gleichen Reihenfolge eine Anzahl an Maschinen durchlaufen, auf denen sie bearbeitet werden. Die Fortbewegung kann durch ein kontinuierlich laufendes Förderband geschehen, das mit einer konstanten Geschwindigkeit läuft oder das Band wird für die Taktzeit angehalten und anschließend werden alle zu bearbeitenden Produkte an eine nachfolgende Station befördert.

Eine primäre Zielsetzung des Line Balancings von getakteten Linien ist es, die Produktionsprozesse den Stationen so zuzuordnen, dass die eingesetzten Mitarbeiter in der Linie gleichmäßig und hoch ausgelastet sind. Hierzu gibt es eine Vielzahl an mathematischen Heuristiken und Optimierungsverfahren, die in Abhängigkeit des anliegenden Produktionsprogramms die Austaktung der Stationen unterstützen. In der Literatur finden sich hierzu eine Reihe von Beispielen, die eine Einordnung der Lösungsverfahren für die unterschiedlichen Ausprägungen von getakteten Fließbändern aufzeigen [1, 2].

Im vorliegenden Beitrag wird der Ansatz des simulationsgestützten Line Balancings vorgestellt. Die

Simulation dient hierbei als Bewertungsfunktion der vorab eingestellten Linienkonfiguration. Im Kontext der betrieblichen Praxis können mit Zunahme des Planungshorizonts immer mehr Stellparameter bei der Suche nach einer besseren Konfiguration berücksichtigt werden.

1 Stellgrößen der Fließbandabstimmung

Im Kurzfristbereich bis kurz vor Produktionsstart können Änderungen im Personalantritt hinsichtlich ihrer Auswirkungen auf die Auslastung und Machbarkeit des Produktionsprogramms analysiert und bewertet werden. Als Stellgrößen dient hierbei die Anzahl der zur Verfügung stehenden Mitarbeiter und deren Qualifikation und Rollen (Mitarbeiter an einer Station, Springer, etc.).

Im Zeithorizont von einer Schicht bis zu mehreren Tagen im Voraus sind die Auftragsreihenfolgen festzulegen. Dies setzt den Umfang der in dieser Periode zu produzierenden und spezifizierten Aufträge voraus und erfolgt üblicherweise über eine Sequenzoptimierung. Im Gegensatz zur Ermittlung einer optimierten Sequenz anhand von Dichte- bzw. Abstandsregeln, die die Engpassserfahrungen aus der Vergangenheit auf die Zukunft projizieren, bewertet der simulationsbasierte Ansatz jedes Sequenzszenario sehr präzise hinsichtlich zukünftiger Auslastung und Überlastfälle. In Kombination mit regelbasierten und heuristischen Suchalgorithmen bzw. mit dem Ansatz einer vollständigen Enumeration wird bei beliebigem Produktionsmix eine (nahezu) optimale Produktionsreihenfolge bestimmt.

Der Begriff des Line Balancing wird in den meisten Fällen mit der Aufgabe und Fähigkeit verbunden, sowohl Kapazitäten, Prozesse und Taktzeiten so zu verteilen bzw. zu ändern, dass die o.g. Ziele erreicht werden. Die Kapazitäten können dabei konkrete Personen/ Ressourcen oder auch Platzhalter repräsentieren. In den Arbeitsplänen sind die Prozesszeiten enthalten, die je Prozessschritt einer Station zugeordnet sind. Diese Prozesse können geteilt, anderen Stationen zugeordnet oder auch durch andere Prozesse substituiert werden. Eine

Taktzeitveränderung wirkt sich auf die Auslastung aller Stationen bzw. den zugeordneten Mitarbeitergruppen aus. In der Praxis wird dieser Anwendungsfall auch häufig "Umtaktung" genannt.

Darüber hinaus sind Prozessänderungen mit der Einführung neuer Produktausprägungen zu erwarten. Die Einführung neuer Komponenten fällt zumeist unter dem Begriff der "Vertaktung".

Im strategischen Planungsfall geht es um eine Neu- oder Erweiterungsplanung der Linie, bei der neue Produkte und Änderungen der Linienstruktur betrachtet werden.

In nachfolgender Abbildung sind die aufgezeigten Anwendungsfälle in Abhängigkeit der Planungshorizonte dargestellt. In Abgrenzung zu den Planungsfällen vor dem Produktionsstart dienen Rückmeldungen während der Produktion zur Steuerung des Personaleinsatzes.

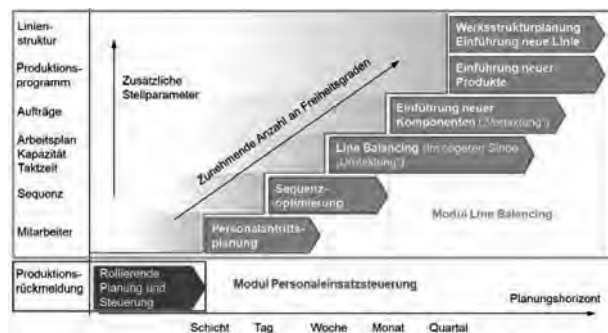


Abbildung 1: Anwendungsfälle und Stellparameter zum Line Balancing

2 Dynamische Wirkzusammenhänge des Personaleinsatzes

Der Einsatz eines Mitarbeiters zur Bearbeitung eines Prozesses an einer Station zu einem bestimmten Zeitpunkt ist von einer Vielzahl an Einflüssen abhängig. Die nachfolgende Auflistung der wesentlichen Einflussgrößen dient zur Illustrierung der Komplexität zur Vorhersage, wie gut ein anliegendes Produktionsprogramm durch die verfügbaren Mitarbeiter bearbeitet werden kann.

2.1 Einflussgröße Auftragssequenz

Die Reihenfolge der Aufträge in der Einsteuerung in die Linie hat einen wesentlichen Einfluss auf die Auslastungsprofile der Mitarbeiter. Aufgrund der Prozesszeitvarianz der Produktausprägungen werden die Mitarbeiter teilweise über- als auch unterausgelastet. Durch

geschickte Wahl der Auftragssequenz können solche Schwankungen über mehrere Aufträge ausgeglichen werden, indem Aufträge mit niedrigen Prozesszeitanforderungen nach Aufträgen mit hohen Prozesszeitanforderungen folgen. Da dies je Auftrag von Station zu Station unterschiedlich sein kann, sind eine Vielzahl an Glättungsverfahren entwickelt worden, um eine über alle Mitarbeitergruppen harmonisierte Auslastung zu erhalten [3, 4, 5, 6].

2.2 Einflussgröße Arbeitspläne und Prozesse

Die Prozesse je Station und Auftrag sind in den Arbeitsplänen hinterlegt. Je nach Ausstattungsvariante des Produkts werden die Prozesse den Stationen zugeordnet. Durch die Festlegung der Position in der Auftragssequenz kann genau ermittelt werden, welche Prozesszeitanforderungen je Station und Takt anliegen.

2.3 Einflussgröße Mitarbeiter

Die Mitarbeiter bearbeiten die Prozesse an den Stationen. Ob ein Mitarbeiter einen Prozess bearbeiten kann, hängt von einer Reihe von Randbedingungen ab, wie beispielsweise

- Zuordnung zu einer oder mehreren Stationen
- Qualifikation
- Anzahl frei verfügbarer Mitarbeiter für den Prozess
- Prozessabhängigkeiten
- Sperrzeiten im Takt
- Driftvermögen (Vorziehen bzw. Nacharbeit)

2.4 Sonstige Einflussgrößen

Darüber hinaus gibt es weitere Parameter, wie z.B. die Taktzeit oder die Struktur der Linie (Anzahl und Anordnung der Stationen), die einen Einfluss auf die Aufgabe des Line Balancings haben.

Alle aufgezeigten Einflussgrößen wirken auf die Auslastung der Mitarbeiter und sind im Kontext der zeitlich wechselnden Anforderungen hinsichtlich der Auswirkungen teilweise schwer prognostizierbar. Um eine exakte Vorhersage des Ablaufs in einer getakteten Linie zu erhalten, bietet sich die ereignis-diskrete Simulation an [7].

Durch die Variation der Stellparameter in Abhängigkeit des Planungshorizonts (Abbildung 1) können unterschiedliche Szenarien gebildet und bewertet werden. In Abbildung 2 sind beispielhaft die Einflussfaktoren und den zugeordneten Planungsrubriken aufgezeigt.



Abbildung 2: Stellparameter zur Szenarienplanung

3 Simulation der Mitarbeiterauslastung

Die ereignis-diskrete Simulation ermöglicht unter Berücksichtigung aller vorab aufgezeigten Einflussgrößen die exakte Vorhersage, wann welcher Mitarbeiter(typ) in welcher Station zu welchem Zeitpunkt wie hoch ausgelastet sein wird. Je nach Anwendungsfall wird bei Engpasssituationen der Mitarbeiter driften, ein Springer eingesetzt oder die Taktzeit temporär für einen Zyklus verlängert.

Die Simulation ermittelt für jeden einzelnen Prozess zum Zeitpunkt des Bedarfstermins die Verfügbarkeit zulässiger Mitarbeiter. Anhand der Verwaltung der Ereignisse über der Zeit ist zu jedem Planungszeitpunkt bekannt, in welchem Zustand (frei oder belegt) sich die Mitarbeiter befinden. Damit werden nicht nur Engpässe transparent, sondern auch deutliche Unterauslastungen.

Durch die Festlegung der Regeln der Zusammenarbeit und der Restriktionen wird ein sehr präzises Bild der Auslastungssituation des simulierten Zeitraums möglich. In Abbildung 3 sind exemplarisch die Parameter aufgezeigt, die möglich und zugleich nötig sind, um Aussagen über die Auslastung valide treffen zu können.

Die hohe Präzision in der Vorhersage des dynamischen Verhaltens in der Linie ist die Grundlage für die valide Bewertung alternativer Szenarien. Erst mit dieser Genauigkeit sind Lösungen auffindbar, die mit einem reinem Kapazitäts-Bedarfsabgleich auf Stations-Taktzeit-Ebene nicht möglich sind. Die weitgehende Überdeckung der Planungsergebnisse mit der Realität ist zwingende Voraussetzung für eine zielorientierte Optimierung (im Sinne von Verbesserung). Somit ist die simulationsbasierte Abbildung der Produktionslinie eine ideale Methode für das Line Balancing.

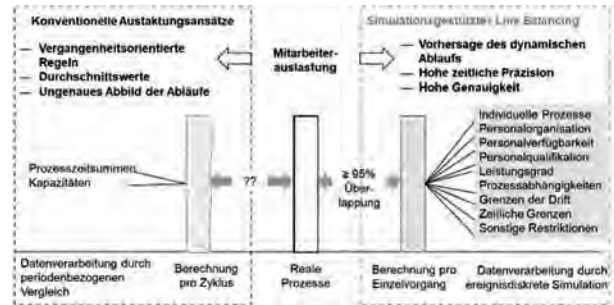


Abbildung 3: Vergleich der berücksichtigten Einflussgrößen für das Line Balancing

4 Simulationsgestütztes Line Balancing

Unter Line Balancing in diesem Beitrag werden alle Tätigkeiten verstanden, die eine Veränderung an einem der aufgezeigten Stellparameter vornimmt. Die Änderungen können an einem oder mehreren Parametern vorgenommen werden. Die Ermittlung der Auswirkungen auf die Mitarbeiterauslastung erfolgt durch die Simulation des Ablaufverhaltens der Linie unter Berücksichtigung der in Abbildung 3 aufgezeigten Einflussgrößen.

Die Bewertung erfolgt anhand von Auslastungskennzahlen. In Abbildung 4 ist beispielhaft die Prozesszeitsummenvarianz zwischen den minimalen und maximalen Prozesszeitsummen je Station und die durchschnittliche Auslastung der Mitarbeitergruppen in dem ausgewählten (Simulations-)zeitraum visualisiert.

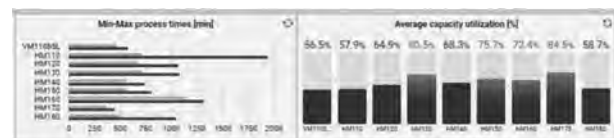


Abbildung 4: Kennzahlen zur Mitarbeiterauslastung

Der Ablauf des Line Balancings erfolgt durch das Wechselspiel von Parameteränderung und Bewertung durch die Simulation. Im Falle einer geänderten Prozesszeitzuordnung von einer Station zu einer anderen wirkt sich diese Änderung auf alle Aufträge aus, die diesen Prozess im Arbeitsplan enthalten haben. Mit der visuellen Unterstützung in der Darstellung von Prozesszeitsteigerungen und der Funktionalität zum Verschieben von Prozessen per Drag&Drop mit anschließender, simulationsbasierter Bewertung der Auswirkungen, wird

eine zielgerichtetes und präzises Line Balancing ermöglicht.



Abbildung 5: Line Balancing durch Zuweisung von Prozessen zu alternativen Stationen

Die Optimierung erfolgt durch die zielgerichtete Änderung von Stellgrößen in einem Szenario und anschließender Simulation und Bewertung. Dies kann durch einen Planer erfolgen oder über einen Algorithmus, der anhand der Simulationsergebnisse auf die nächsten Stellgrößenparametrisierung schliesst. Im einfachsten Fall beschränkt sich die Optimierung auf die Festlegung der Einstreuungsreihenfolge, bei der die Freiheitsgrade in der Sequenzpositionierung der Aufträge limitiert sind. Im Falle von Prozessverschiebungen sind weitergehende Informationen über Vorgänger-Nachfolger-Beziehungen zu hinterlegen. Da hierzu ein tiefergehendes Verständnis über die technischen Vorgänge notwendig ist, ist eine manuelle Verschiebung durch einen Planer einer automatisierten Lösungssuche vorzuziehen.

5 Fazit und Ausblick

Im Beitrag wird anhand eines Praxisbeispiels aufgezeigt, wie die Fließbandabstimmung von der operativen bis zur mittel- bis langfristigen Planung in einer Line Balancing-Applikation vorgenommen werden kann. Die Stellparameter werden hinsichtlich ihrer dynamischen Wirkzusammenhänge und ihr Zusammenhang mit den Anwendungsfällen aufgezeigt.

Das simulationsbasierte Line Balancing bietet große Vorteile. Insbesondere durch kurze Simulationslaufzeiten sind dialogfähige Planungsverbesserungen in Kombination mit dem Know-how der Planer realisierbar. Dies wird die Art und Häufigkeit von Fließbandabstimmungen verändern. So werden beispielsweise Taktzeitänderungen nicht mehr aufwändig durchgerechnet. Anstatt alle paar Monate zwischen vordefinierten Taktzeitmodellen zu wechseln, werden zukünftig die Taktzeiten an das

bestehende Produktionsprogramm angepasst und somit die Produktivität hochgehalten. Anhand von Live-Demos werden Beispiele aus der Praxis demonstriert.

Referenzen

- [1] Boysen N., Fliedner, M., Scholl, A. *A classification of assembly line balancing problems*. European Journal of Operational Research. Volume 183, Issue 2, 1 December 2007, p. 674-693
- [2] Naveen K., Dalgobind M. *Assembly Line Balancing: A Review of Developments and Trends in Approach to Industrial Application*. In: Global Journal of Researches in Engineering Industrial Engineering. Volume 13 Issue 2 Version 1.0 Year 2013. Global Journals Inc. (USA)
- [3] Nuchsa K., Pianthong N. *The Assembly Line Balancing Problem: Review articles*. KKU Engineering Journal Vol. 34 No .2 (133 - 140) March – April 2007 p 133 - 140.
- [4] Scholl A., Becker C. *State of the art exact and Heuristic solution procedures for simple assembly line balancing*. European Journal of Operations Research. 2011. 168(3): p. 666-693.
- [5] Sivasankaran P, Shahabudeen, P. *Literature review of assembly line balancing problems*. Int J Adv Manuf Technol (2014) 73. p. 1665–1694
- [6] Belfiore, G., Falcone D., Silvestri, L. *Assembly line balancing techniques: literature review of deterministic and stochastic methodologies*. Conference: The international conference on modelling and applied simulation Budapest 2018
- [7] Pröpster, M.; März, L.; Reinhart, G.; Intra, C.: *Validation of Line Balancing by Simulation of Workforce Flexibility*. 9th CIRP Conference on Intelligent Computation in Manufacturing Engineering (ICME 2014). Capri, 24.07.2014.

Simulating the Viability of Crowd Logistics in Short Food Supply Chains

Florian Cramer*, Christian Fikar

Chair of Food Supply Chain Management, Faculty of Life Sciences: Food, Nutrition and Health, University of Bayreuth, Fritz-Hornschuch-Straße 13, 95326 Kulmbach, Germany

*florian.cramer@uni-bayreuth.de

Abstract. This work presents a simulation model to evaluate the viability of crowd logistics concepts in context of short food supply chains (SFSC). Many local agri-food supply chains, especially those of smallholder farmers, lack efficient and sustainable means for physical distribution. By employing a hybrid approach, based on agent-based and discrete event simulation modelling, a decision support system is developed to help decision makers to estimate the effects of crowd logistics on farmers' market reach, vehicle kilometres travelled, and food quality in SFSC. Preliminary results show that crowd logistics can increase market reach while only slightly increasing vehicle kilometres travelled, highlighting that crowd logistics can be a viable option for last-mile distribution in SFSC.

Introduction

Regional and local development has been acknowledged as a driver of sustainability efforts in food supply chains [1]. Supporting local smallholder farmers has the advantage of facilitating development of local communities and local employment opportunities, moreover, biodiversity can be enhanced. One particular challenging problem in short food supply chains (SFSC) is a sustainable, physical distribution [2]. Moreover, a growing complexity of dietary needs, concerns about food quality, and diet choices of modern consumers bundled with a preference for convenient shopping experiences complicate and limit the distribution of locally sourced and grown agri-food products [3, 4]. While consumers might have to visit different outlets to satisfy their needs when shopping at alternative food outlets, many consumers might actually not shop at these outlets due to a lack of shopping convenience. One logistics concept that is rather unexplored in regards to agri-food supply chains is crowd logistics. Crowd logistics takes advantage of pre-existent routes and enables transportation by hitchhiking unused transportation capacities [5]. Thus, crowd logistics is able to save vehicle kilometres travelled (VKT) by reducing the

need of dedicated transport routes. According to Paciarotti and Torregiani [6], services based on IT-platforms have not yet been properly developed for SFSCs. Thus, this research aims to provide a decision support system (DSS) to explore crowd logistics in the context of alternative food networks (AFN) and SFSCs. The DSS is based upon a combination of elements of agent-based and discrete event simulation modelling, thus using a hybrid approach. It is aimed at decision makers to provide a better understanding of how crowd logistics could influence smallholder farmers market reach, its impact on food quality and VKT. Consequently, the aim of this work is to provide a simulation model that reflects the viability of crowd logistics for last-mile delivery in SFSC, by exploring different variations of the crowd logistics transportation concept in SFSC.

1 Related Work

Even though simulation modelling holds a great potential to facilitate sustainable operations in SFSC [7], only few research is found in this area. The use of crowd logistics in this regards remains mostly unexplored. Fikar and Leithner [7] use a DSS based on a discrete event simulation modelling and an adaptive large neighbourhood search metaheuristic to examine whether logistics collaboration can improve SFSC efficiency. Their results indicate that collaboration in SFSC has a high potential to reduce vehicle kilometres travelled and the amount of required vehicles for transport. However, at the same time they discovered that this reduction comes at the cost of reduced food quality. Mittal and Krejci [8] use a hybrid approach, combining agent-based and discrete event simulation modelling, to create a simulation framework that allows decision makers to explore how the logistical and operational efficiencies of food hubs in SFSC could be increased. The authors

show that while financial incentives to facilitate producer cooperation in regard to delivery scheduling are less likely to improve food hub efficiency, adaptive personnel scheduling can be used to improve efficiency without interfering with producer interests. Mittal et al. [9] explore the use of crowd logistics for food rescue programs by developing an agent-based simulation. Their model demonstrates that crowd logistics has the potential to be a viable option for physical distribution in such settings. Melkonyan et al. [2] research the potential effects on sustainability of different distribution strategies in SFSCs, by employing a combination of system dynamics simulation modelling and multi-criteria decision analysis. The results show that crowd logistics has the potential to increase SFSC sustainability. In contrast to their research, this work focuses solely on the impact of crowd logistics as means of physical distribution in SFSC. Furthermore, the simulation developed in this paper also considers potential effects on food quality.

2 Hybrid Simulation

Hybrid simulation is the combination of multiple simulation modelling techniques, mostly a combination of either system dynamics, discrete event, and agent-based simulation modelling [10]. Agent-based simulation modelling (ABS) allows to model individual unit (agents) to interact with other agents and the environment as individuals with independent characteristics that act autonomously whose behaviour is controlled by stochastics [11]. In discrete event simulation modelling (DES) entities move through networks of queues and activities using available shared resources where applicable [12]. The model developed for this research was build in accordance to the iterative modelling process as described by Law [13]. A combination of agent-based and discrete event simulation modelling was chosen to model cooperative and competitive behaviour of crowd logistics participants, more commonly referred to as occasional drivers (OD), and behavioural aspects such as scheduling or shopping decisions. For the experiments both urban and rural infrastructure is considered. For the study areas, the regions of Upper Franconia, Germany was chose to represent a rural setting while Munich Municipality, Germany, was chosen for the urban one. As for the simulation duration, a timeframe of three month was chosen to reflect the time period (June to September) where according to local authorities (cf.

[14]), most fresh food in the simulated areas is in season. A total of three scenarios were developed for the simulation (cf. Figure 1): (i) A base line scenario which does not include any crowd logistics activities, (ii) a basic crowd logistics scenario in which shoppers at alternative food outlets may participate as ODs, and (iii) an advanced crowd logistics scenario where shoppers and non-shoppers may participate as ODs.

The structure of the DSS is depicted in Figure 2. The simulated areas consist of a number of outlets and populated areas. Each populated area contains a number of households, which in return contain a stochastically assigned number of inhabitants. The food demand is handled on the household level, while any activities such as working, shopping, or participating as OD is handled by the individual inhabitants. Each inhabitant has its own schedule, detour willingness and maximum detour distance and willingness to participate as OD. To model loss of food quality, the keeping quality model by Tijssens and Polderdijk [15] is used. The outlets generate baskets with stochastically assigned properties (to simulate cross-perishability and differentiating food sensitivity) upon serving a customer. When delivered, the baskets may be stored at a household's doorstep at ambient temperature until pickup by an inhabitant. Each baskets conditions in regard to food quality and travelled route are evaluated after pickup at the destination household's doorstep.

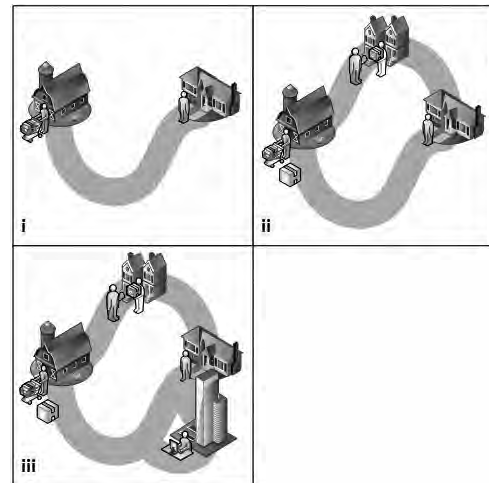


Figure 1: Three scenarios are explored in both, an urban and a rural setting, including one base line scenario and two crowd logistics scenarios.

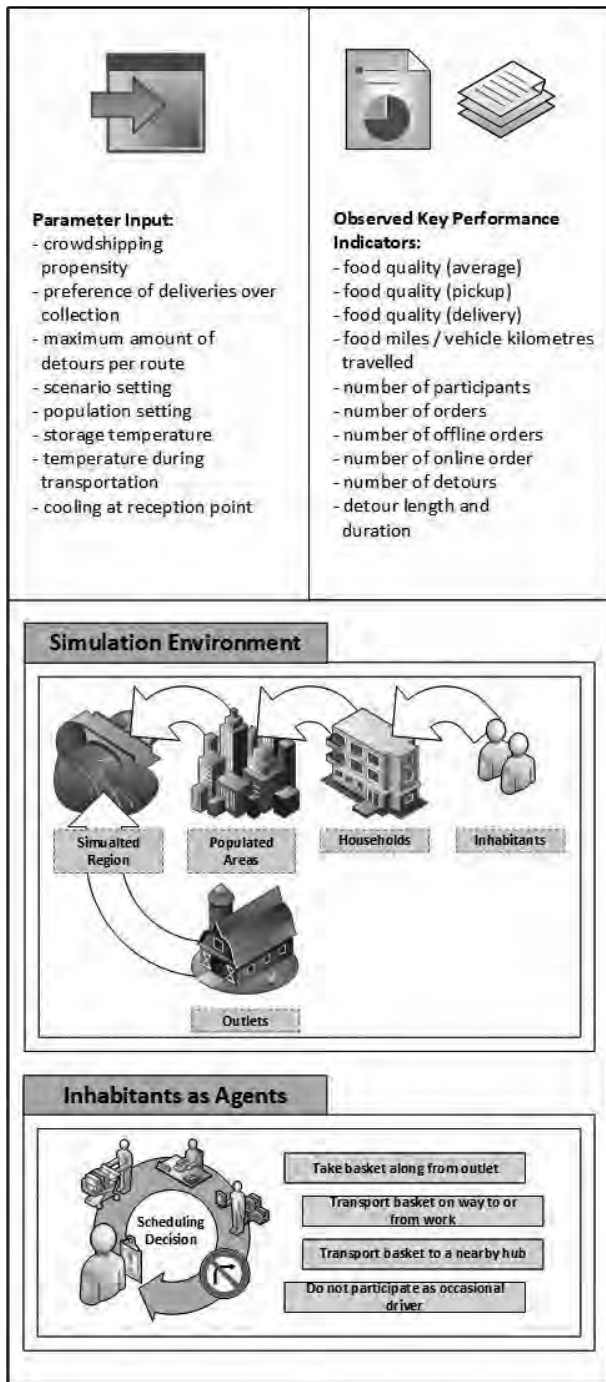


Figure 2: The DSS to estimate the viability of the crowd logistics concept.

3 Preliminary Results

A preliminary analysis, using 10 replications for each simulation run was carried out. All three scenarios

were examined for an urban population setting. For Scenario 2 and Scenario 3, a propensity to participate as OD between 10% and 100% was simulated with 10% intervals. The parameter for preference of deliveries over collection was set to 0%. The first scenario, the base line scenario, was used as reference point for the observations. With the given parameters, the analysis indicated that the effect of employing crowd logistics depends on available infrastructure, with a higher effectiveness in rural areas. Furthermore, according to preliminary results for an urban setting, crowd logistics could increase the reach of smallholder farmers even at relatively low rates of participation.

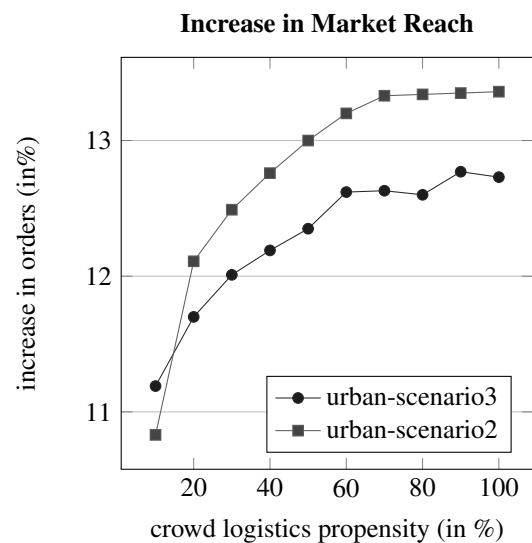


Figure 3: Increase in market reach in % of gained orders dependent on ODs crowd logistics propensity.

The lower effectiveness of also employing non-shoppers as ODs (Scenario 3), can be attributed to the higher competition over orders, which may lead to a sub-optimal allocation. Figure 4 shows, that at the same time Scenario 3 leads to a lower increase of VKT per basket.

4 Conclusion

This work introduced a simulation-based DSS to investigate the viability of using crowd logistics concepts for physical transportation in SFSC. The preliminary results show that crowd logistics could be a viable option for last-mile delivery in SFSC, even with a relatively

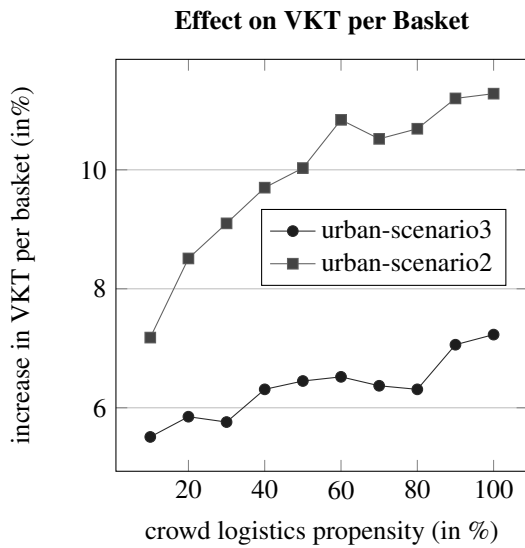


Figure 4: Increase in VKT per shopping basket dependent on ODs crowd logistics propensity.

low amount of participants. Furthermore, the results indicate that the developed DSS can be a useful tool for decision makers to facilitate more sustainable food systems. Further work will include a more sophisticated analysis and computational experiments with a larger number of replications as well as further model validation.

References

- [1] Lamine C, Garçon L, Brunori G. Territorial agrifood systems: A Franco-Italian contribution to the debates over alternative food networks in rural areas. *Journal of Rural Studies*. 2019;68(3):159–170.
- [2] Melkonyan A, Gruchmann T, Lohmar F, Kamath V, Spinler S. Sustainability assessment of last-mile logistics and distribution strategies: The case of local food networks. *International Journal of Production Economics*. 2020;228(2):107746.
- [3] Leone LA, Fleischhacker S, Anderson-Steeves B, Harper K, Winkler M, Racine E, Baquero B, Gittelsohn J. Healthy Food Retail during the COVID-19 Pandemic: Challenges and Future Directions. *International journal of environmental research and public health*. 2020; 17(20).
- [4] Wang O, Somogyi S, Charlebois S. Food choice in the e-commerce era. *British Food Journal*. 2020; 122(4):1215–1237.
- [5] Buldeo Rai H, Verlinde S, Macharis C. Shipping outside the box. Environmental impact and stakeholder analysis of a crowd logistics platform in Belgium. *Journal of Cleaner Production*. 2018;202(2):806–816.
- [6] Paciarotti C, Torregiani F. The logistics of the short food supply chain: A literature review. *Sustainable Production and Consumption*. 2021;26(2):428–442.
- [7] Fikar C, Leithner M. A decision support system to facilitate collaborative supply of food cooperatives. *Production Planning & Control*. 2021; 32(14):1179–1190.
- [8] Mittal A, Krejci CC. A hybrid simulation modeling framework for regional food hubs. *Journal of Simulation*. 2019;13(1):28–43.
- [9] Mittal A, Oran Gibson N, Krejci CC, Marusak AA. Crowd-shipping for urban food rescue logistics. *International Journal of Physical Distribution & Logistics Management*. 2021;51(5):486–507.
- [10] Brailsford SC, Eldabi T, Kunc M, Mustafee N, Osorio AF. Hybrid simulation modelling in operational research: A state-of-the-art review. *European Journal of Operational Research*. 2019;278(3):721–737.
- [11] Macal CM. Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*. 2016;10(2):144–156.
- [12] Robinson S. Discrete-event simulation: A primer. In: *Discrete-Event Simulation and System Dynamics for Management Decision Making*, edited by Brailsford S, Churilov L, Dangerfield B, vol. 2, pp. 10–25. Chichester, UK: John Wiley & Sons Ltd. 2014;.
- [13] Law AM. *Simulation Modeling and Analysis*. New York: McGraw-Hill US Higher Ed USE Legacy, fifth edition ed. 2015.
- [14] Kern. Wegweiser Gemeinschaftsverpflegung. Bayerischer Saisonkalender [Guideline to Mass Catering. Bavarian Seasonal Calendar]. 2020. URL https://www.kern.bayern.de/mam/cms03/shop/flyer/dateien/200407_final_kern_saisonkalender_barrierefrei_neu.pdf, 31.03.2022
- [15] Tijskens L, Polderdijk JJ. A generic model for keeping quality of vegetable produce during storage and distribution. *Agricultural Systems*. 1996;51(4):431–452.

Kopplung einer Material- und Energieflusssimulation mit Reinforcement-Learning-Algorithmen

Johannes Dettelbacher*, David Wagner*, Alexander Buchele, Wolfgang Schlüter

Hochschule Ansbach, Residenzstraße 8, 91522 Ansbach,

**Die Autoren haben gleichberechtigt zur Veröffentlichung beigetragen*

Abstract. Das Thema industrielle Energieeffizienz nimmt, gerade im Hinblick auf die Energiewende, weiter an Bedeutung zu. Es gilt dabei den Ausstoß von Treibhausgasen zu minimieren bei gleichbleibender oder sogar gesteigerter Produktivität. Die mathematische Modellierung bietet dabei durch die Möglichkeit von Simulations- und Optimierungsmethoden Potentiale den Betriebsablauf hinsichtlich dieser Ziele zu verbessern. Im Rahmen dieser Veröffentlichung wird ein Reinforcement Learning Algorithmus verwendet, der in der Lage ist, optimierte Betriebszustände in einer hybriden Material- und Energieflusssimulation zu erreichen. Die so erarbeitete Methode findet beispielhaft auf einem Schmelzbetrieb Anwendung und ermittelt in diesem Zusammenhang die optimale Beschickung von Schmelzöfen. Erste Ergebnisse zeigen, dass Maschinenstillstände durch die Anwendung vermieden werden können.

Einleitung

Im Zusammenhang mit dem Ziel der Energiewende ergeben sich für viele Industriebetriebe in Deutschland enorme Herausforderungen. Insbesondere durch steigende Energiepreise rückt Energieeffizienz im industriellen Umfeld in den Fokus. Neben technischen Maßnahmen, welche bereits zahlreich umgesetzt werden, bieten Optimierungen im Betriebsablauf zusätzliche Potentiale. Um diese zu nutzen, können Werkzeuge wie die Simulation und mathematische Optimierung eingesetzt werden. Während die Simulation das dynamische Verhalten hinsichtlich Material- und Energiefluss betrachtet, können mathematische Optimierungsmethoden die Einflussparameter für einen verbesserten Betriebszustand identifizieren. Während der Energiefluss jedoch über ein kontinuierliches Wärmetransport-Modell beschrieben werden kann, erfolgt die Materialflusssimulation ereignisdiskret.

Die Kopplung beider Systeme wird daher mittels

hybrider Modelle realisiert. Im Rahmen dieser Arbeit werden die Möglichkeiten einer Kopplung zwischen hybrider Material- und Energieflusssimulation und Optimierungsmethoden weiter untersucht. Speziell wird dabei auf die mögliche Verknüpfung der Simulation mit der Methode des Reinforcement Learning eingegangen.

1 Hybride Material- und Energieflusssimulation

In realen Prozessen betrachtet man häufig hybride Systeme, welche sich durch die Verknüpfung mehrerer Simulationsmethoden beschreiben lassen. Während der Materialfluss in der Regel über eine ereignisdiskrete Simulation beschrieben wird, kann der Energiefluss über eine kontinuierliche Simulation abgebildet werden. Bei der bidirektionalen Kopplung der einzelnen Simulationsmodelle müssen die Daten, wie in Abbildung 1 gezeigt wird, zwischen den Modellen in jedem Zeitschritt ausgetauscht werden.

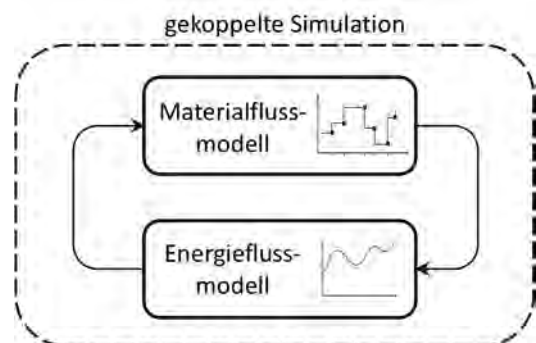


Abbildung 1: Hybride Material- und Energieflusssimulation.

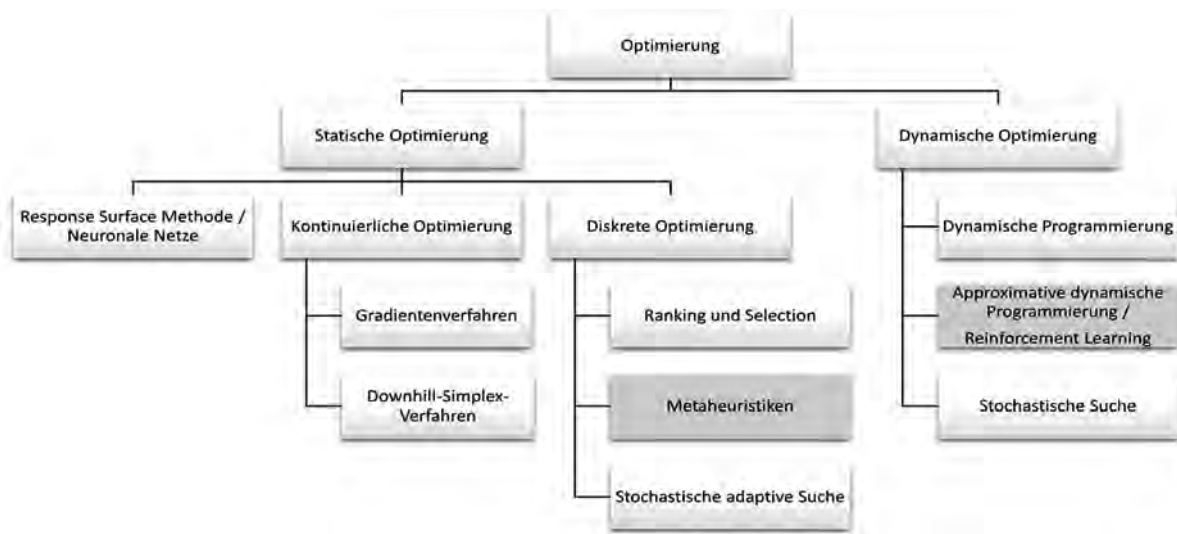


Abbildung 2: Methoden für die simulationsbasierte Optimierung [4], [5].

Für die Kopplung dieser Simulationsparadigmen sind bereits Methoden in der Literatur beschrieben. Peter und Heinzl beschreiben jeweils Möglichkeiten zur Verknüpfung beider Simulationsmethoden [1], [2]. So können für eine hybride Simulation die Subsysteme (ereignisdiskret und kontinuierlich) in einer Simulationsumgebung sowie mehreren Simulationsumgebungen abgehandelt werden (Co-Simulation).

2 Simulationsbasierte Optimierung

Um mithilfe des hybriden Modells optimierte Betriebszustände zu ermitteln, kann eine simulationsbasierte Optimierung nach [3] verwendet werden. Eine Auflistung möglicher Optimierungsmethoden, welche für simulationsbasierte Optimierungen eingesetzt werden können, ist in **Fehler! Verweisquelle konnte nicht gefunden werden.** dargestellt. Aufgrund der dynamischen Simulation kann eine dynamische Optimierung vorgenommen werden. Die getroffenen Aktionen im Materialfluss bedingen neue Zustände in der hybriden Simulation. Sie beschränken damit zukünftige Aktions- und Zustandsräume. Es muss daher in jedem Zeitschritt ein neues Optimum berechnet werden, welches auch im zeitlichen Verlauf optimale Ergebnisse erzielt. Die Simulation ist dafür in die Optimierung selbst eingebettet und wird fortlaufend mit optimierten Parameterwerten ausgeführt.

In der Literatur sind bereits Methoden für die

Verknüpfung von hybriden Simulationen mit statischen Optimierungsmethoden, wie Metaheuristiken, beschrieben. Sobottka und Römer stellen hier Methodiken vor, welche sich insbesondere für die Produktionsplanung eignen [6], [7] und auf einer statischen Optimierung basieren. Die dynamischen Optimierungsmethoden können aufgrund ihrer geringen Rechendauer bei der Verwendung des zuvor trainierten Agenten in neuen Einsatzgebieten wie der Produktionssteuerung Anwendung finden. In Abbildung 3 wird die Optimierung mittels Metaheuristiken dem Reinforcement Learning gegenübergestellt. Beide Methoden können nahezu optimale Lösungen finden. Während jedoch bei der Anwendung von Metaheuristiken die möglichen Eingänge direkt über bspw. genetische Algorithmen bestimmt werden, wird im Reinforcement Learning ein Agent sukzessive angepasst.

Damit liegt der Aufwand beim Reinforcement Learning vor dem Einsatz des Agenten. Die Implementierung des Simulationsmodells für den Anwendungsfall ist in beiden Fällen identisch. Dafür erfordert jedoch die Umsetzung der Optimierungstechnik beim Reinforcement Learning sowie die Schnittstellen zum Simulationsmodell Prozessverständnis, aufgrund der notwendigen Anpassung der Hyperparameter und die Auswahl der Zustandsgrößen. Weiterhin verlangen die Selektion eines geeigneten Agenten sowie die Einstellung der Parameter sowohl Kenntnisse des Optimierungsproblems selbst als auch der Optimierungsmethode.

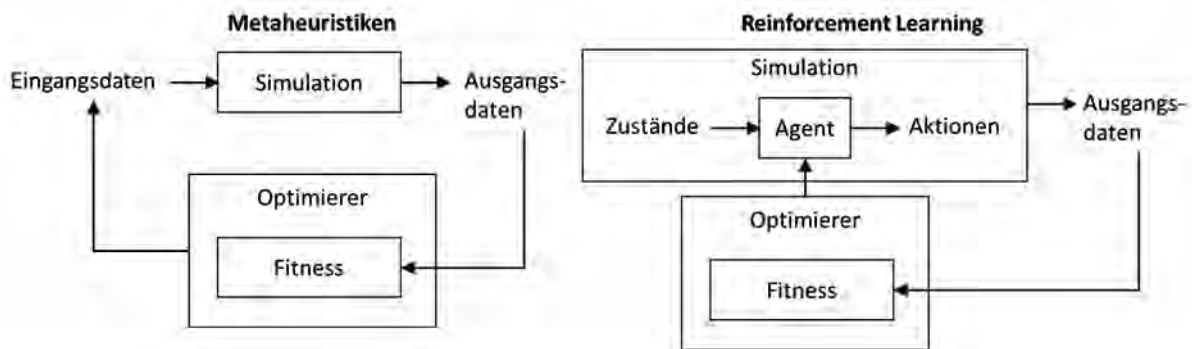


Abbildung 3: Simulationsbasierte Optimierung mit Metaheuristiken (links) und Reinforcement Learning (rechts) [8].

3 Reinforcement Learning

Die Verknüpfung der hybriden Simulation mit dem Reinforcement Learning verfolgt den in Abbildung 3 gezeigten Ansatz. Eine Erläuterung der einzelnen Komponenten hinsichtlich der Methodik findet sich in Tabelle 1.

Begriff	Begriffserläuterung
Simulation	Material- und Energieflusssimulation
Zustände	Kontinuierliche und diskrete Zustandsgrößen
Agent	Bewertung der Aktionen für jeweilige Zustände (z. B. Neuronales Netz)
Aktionen	Entscheidungsvariablen für das System
Ausgangsdaten	Ergebnisse aus Simulationsmodell (z. B. Produktivität und Energieverbräuche)
Kostenfunktion	Kostenfunktion aus gewichteten (diskreten und kontinuierlichen) Ausgangsdaten
Optimierer	Optimierer für die Anpassung des Agenten

Tabelle 1: Beschreibung der Komponenten aus dem Reinforcement Learning in Abbildung 3.

Der Ansatz des Reinforcement Learning wird bereits für die Optimierung des Produktionsablaufes genutzt [9]. Dabei wird vornehmlich die Kopplung der Optimierungsalgorithmen mit ereignisdiskreten Simulationsmodellen im Vordergrund. Die Kopplung mit hybriden Simulationsmodellen ist jedoch in der Literatur unterrepräsentiert und erfordert weitergehende Untersuchungen. Im Folgenden werden die Komponenten für die betrachtete Anwendung erläutert. Die hybride Material- und Energieflusssimulation stellt die Umgebung für den Agenten dar.

Der Agent hat dabei die Aufgabe in einer bestimmten Frequenz mit der Simulationsumgebung zu interagieren. Als Entscheidungsgrundlage können sowohl kontinuierliche als auch diskrete Zustandsgrößen aus der Simulation an den Agenten übergeben werden. Diese werden auf Basis einer zuvor festgelegten Bewertungs-/Zielfunktion evaluiert. Der Agent ermittelt für die Zustände des Systems geeignete Folgeaktionen. Hierfür wird die Netzwerkarchitektur des Agenten, welcher über ein neuronales Netz abgebildet werden kann, so angepasst, dass die getroffenen Entscheidungen zum Maximieren der Zielfunktion führen. Die Zielfunktion umfasst relevante Ausgangsparameter, welche dem Agenten aus der Simulation zur Verfügung gestellt werden. Im Trainingsschritt werden mehrere Simulationsdurchläufe verwendet, in welcher der Agent die Umgebung erkundet und seine Struktur dementsprechend anpasst. Nach Abschluss des Trainings kann der Agent als Steuerung für die Umgebung verwendet werden und somit die Erkenntnisse aus der Erkundungsphase nutzen.

4 Beispielanwendung

Die beschriebene Methodik wird im Rahmen einer aktuellen Forschungsarbeit auf den Einsatz im Aluminium-Schmelzbetrieb getestet. Der Schmelzbetrieb ist für die Versorgung von produzierenden Gussmaschinen mit flüssigem Aluminium zuständig. Insbesondere die Auslastung und damit einhergehend die Zeitpunkte der Befüllungen der Schmelzöfen mit neuem Material sind entscheidend für die Versorgung der Gussmaschinen mit flüssigem Aluminium und somit für eine energieeffiziente Produktion. Für die Untersuchung wird ein Modell eines Schmelzbetriebes verwendet, welches den dis-

kreten Materialfluss der Beschickung und Entnahme sowie den Material- und Energiefluss in den Schmelzöfen abbildet [10]. Der Schmelzbetrieb, der über eine hybride Material- und Energieflusssimulation abgebildet wird, dient als Umgebung für den Agenten. Als Zustände werden die Füllstände, Status der Maschinen sowie Kennzahlen über die Materialnachfrage übergeben. Der Agent, der auf einem mehrschichtigen neuronalen Netz basiert, muss aus den verfügbaren Schmelzöfen die optionale Beschickung eines Ofens als Aktion auswählen. Als Rückkopplung und damit Bewertung der vorgenommenen Aktionen erhält der Agent ein Feedback, das aus einer multikriteriellen Zielfunktion berechnet wird. In der Funktion werden auftretende Maschinenstörungen sowie Energieverbräuche kombiniert bewertet. Eine beispielhafte Umsetzung der hybriden Simulation erfolgt in Matlab/Simulink, wie in Abbildung 4 dargestellt. Dadurch wird die Simulation der Subsysteme innerhalb einer Simulationsumgebung ermöglicht. Aktuelle Ergebnisse zeigen, dass der Agent so zur Steuerung beiträgt, dass Maschinenstillstände vermieden werden. Jedoch bietet diese Steuerung noch Potential für eine Steigerung der Energieeffizienz. Im Rahmen weiterer Untersuchungen werden verschiedene Agenten getestet und verglichen sowie deren Hyperparameter über Gittersuchen analysiert. Auch die Ausgestaltung der Aktionen, Zustände sowie der Form der Zielfunktion sollen im Rahmen weiterer Arbeiten untersucht werden.

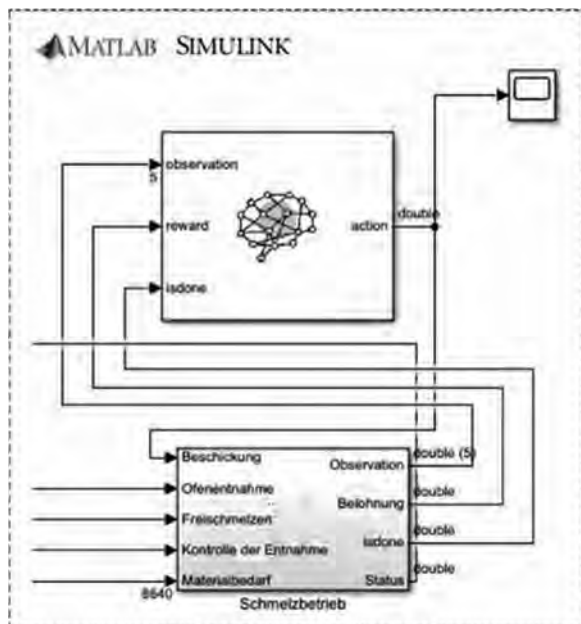


Abbildung 4: Einsatz der Methodik in Matlab/Simulink.

Literatur

- [1] Heinzl B., *Methods for Hybrid Modeling and Simulation-Based Optimization in Energy-Aware Production Planning*, 2020.
- [2] Peter T. *Gekoppelte hybride Simulation von Materialfluss und Energie in Produktion und Logistik*, Shaker Verlag, 2019.
- [3] Verein Deutscher Ingenieure, *VDI Richtlinie 3633 Blatt 12*, 2014.
- [4] Gosavi A. *Simulation based optimization*. 2015.
- [5] Papageorgiou M. *Optimierung: Statische, Dynamische, Stochastische Verfahren*. 2012.
- [6] Sobottka T. *Eine anwendungsorientierte simulationsbasierte Methode, unter Berücksichtigung von Energieeffizienz, in der optimierenden Planung von Produktion und Logistik*, 2017.
- [7] Roemer A., Strassburger S. *A review of literature on simulation-based optimization of the energy efficiency in production*, Winter Simul. Conf., S. 1416–1427, 2016.
- [8] Meyer-Nieberg S., Leopold N., Uhlig T. *Natural Computing for Simulation-Based Optimization and Beyond*. 2020.
- [9] Kuhnle A. et al. *Designing an adaptive production control system using reinforcement learning*, J. Intell. Manuf., S.855-876, 2020.
- [10] Dettelbacher J., Schlüter W. *Material- und Energieflusssimulation zur prädiktiven Bestimmung von Ofenreinigungsintervallen*, Simul. Produktion und Logistik 2021, S. 85–92, 2021.

Erstellung eines Verkehrssimulationsmodells für Berlin in SUMO zur Simulation der möglichen Emissionen und daraus resultierendem Routing

Martin Engelmann^{1*}, Jochen Wittmann^{1**}, Volker Wohlgemuth^{1***}

¹ HTW Berlin, Wilhelminenhofstraße 75A, 12459 Berlin; *martin.engelmann@htw-berlin.de; **jochen.wittmann@htw-berlin.de; ***volker.wohlgemuth@htw-berlin.de;

Abstract. In diesem Projekt wurde mit Hilfe der Daten aus den Verkehrszählungen 2014 ein Verkehrssimulationsmodell für die Stadt Berlin erzeugt. Nach der Erstellung des Modells können die Emissionsfaktoren nach Vorgabe des Handbuchs für Emissionsfaktoren (HBEFA) ausgelesen werden und somit eine Schadstoffermittlung für die simulierten Fahrzeuge ermittelt werden.

Einführung

Durch immer effizientere Verbrennungsmotoren und eine fortschreitende Verbesserung der Abgasreinigung in Kraftfahrzeugen sind die spezifischen Emissionen an Schadstoffen und am Treibhausgas Kohlenstoffdioxid pro Personen- bzw. Tonnenkilometer in Deutschland zum Basisjahr 1995 deutlich gesunken [1]. Im Zeitraum von 1991 bis 2016 stieg die Fahrleistung in Deutschland im Personenverkehr um 31 Prozent sowie im Güterverkehr um 71 Prozent [2]. Die durch technische Maßnahmen erzielten Emissionsminderungen werden durch den Anstieg der absoluten Fahrleistung in Deutschland zum Teil wieder aufgehoben [3]. Die Umweltbelastung durch Kraftfahrzeuge kann alleine durch technische Verbesserungen der Kraftfahrzeuge nicht ausreichend gemindert werden, um einen signifikanten Beitrag zu den Klimaschutzziele Deutschlands zu leisten [4]. Verschiedene Maßnahmen zur Minderung der verkehrsbedingten Emissionen wurden vom Umweltbundesamt in [4] diskutiert. Der Trend in der Antriebstechnik von Fahrzeugen geht derzeit in Richtung Elektromobilität [5]. Dies wird insbesondere durch das gegenwärtige Ziel der Bundesregierung, Deutschland zu einem Leitmarkt für das Thema Elektromobilität zu machen, deutlich [6]. Zur Reduzierung der fahrzeugspezifischen Emissionen ist eine Umstellung der Fahrzeugantriebe auf vollelektrisch oder basierend auf Brennstoffzellen langfristig die beste Lösung

[5]. Kurzfristig wäre eine emissionsoptimierte Routenplanung zur Reduzierung der fahrzeugbedingten Emissionen eine alternative Lösung. In einem vorherigen Projekt wurde zunächst die theoretische Machbarkeit der Routenführung mit Hilfe von Graphhopper gezeigt [7]. Um die dort verwendeten, theoretischen Daten zum Verkehrsaufkommen zu verifizieren, soll nun mit Hilfe einer Simulationssoftware eine möglichst genaue Verkehrsauslastung in einer Stadt, in diesem Fall Berlin, simuliert werden, um dann vor diesem Hintergrund-Verkehr eine emissionsoptimierte Routenführung durchzuführen. Dazu muss zunächst ein Verkehrsmodell für die Stadt Berlin nachgebildet werden. Durch vorherige Projekte konnte gezeigt werden, dass eine emissionsbasierte Routenführung Auswirkungen auf die gewählte Routenführung haben kann [7]. Da der Ansatz bisher nur theoretisch erfolgte, war eine Weiterführung als Simulation angedacht.

Das Projekt wurde in vier Phasen eingeteilt, die aufeinander aufbauend, am Ende die emissionsbasierte Routensimulation erlauben. In dieser Veröffentlichung geht es um die Phasen eins bis drei. Phase Eins beschäftigt sich mit der Erstellung des Verkehrsgraphen bzw. des Verkehrsnetzes sowie das Durchführen von randomisierten Fahrten in der Verkehrssimulationssoftware Simulation of Urban MObility (SUMO). Phase Zwei implementiert die Simulation der Emissionsdaten für alle Fahrten nach HBEFA 3.3 Standard. Sind diese Phasen abgeschlossen wird in Phase drei mit Hilfe des Traffic Control Interfaces (TraCI) Verkehr auf vordefinierten Routen erzeugt, um den Stadtverkehr nachzubilden. In der finalen Phase wird dann in die laufende Verkehrssimulation ein Fahrzeug geladen, das eine Route von A nach B finden soll und dabei möglichst wenig Emissionen erzeugen soll.

1 Datenlage

Das Land Berlin bietet über einen Umweltatlas unterschiedliche Statistiken an. Unter anderem auch ein Verkehrsmodell, welches eine Gesamtverkehrsprognose für Berlin bis 2025 bzw. 2030 bietet. Leider sind diese Daten nicht öffentlich zugänglich. Daher wurden für die vorliegenden Untersuchungen die Ergebnisse der Straßenverkehrszählungen 2014 verwendet [8].

Unabhängig von der Datenlage beschäftigen sich auch andere Projekte mit ähnlichen Aufgabenstellungen. Zum Beispiel zielt das MaLog-Projekt darauf ab, Logistikunternehmen ein web-basiertes Tool zur Verfügung zu stellen um ihre innerstädtische Tourenplanung zu verbessern [11]. Der vorliegende Ansatz behandelt die Aufgabe unabhängig von einer konkreten Anwendungsdomäne als allgemeines Routing-Problem.

1.1 Straßenverkehrszählungen 2014

Zunächst wurden aus dem Ergebnisbericht der Straßenverkehrszählungen 2014 Verkehrsstärkenkarten gebildet. Außerdem wurden die Strecken mit der höchsten Verkehrsstärke erfasst. Daraus ergibt sich die durchschnittliche tägliche Verkehrsstärke der Kraftfahrzeuge (DTV_{WT} KFZ), mit deren Hilfe es möglich war, für einzelne Straßen bzw. Routen Auslastungen sowie eine geschätzte Anzahl der Fahrzeuge zu errechnen.

1.2 HBEFA Implementation in SUMO

Im verwendeten Simulator SUMO sind die Fahrzeug-Emissionsklassen nach HBEFA 2 bzw. HBEFA 3 Standard implementiert. Die Implementation unterscheidet sich im Wesentlichen dadurch, dass die Implementation in Version 3 direkt auf die Klassen des HBEFA erfolgt. [10] Die Unterscheidung der Emissions-Klassen in SUMO gehen nach: *heavy duty emission classes* (HDV), *passenger and light delivery emission classes* (PC bzw. LDV). Zusätzlich wird die Treibstoffart unterschieden mit Diesel (D) und Benzin (G). Sofern bekannt, kann außerdem die Schadstoffklasse nach EU-Norm angegeben werden, dadurch ergibt sich beispielsweise eine Klasse *PC_G_EU4* für ein von einer privat Person betriebenes Benzin KFZ der EU-Norm 4. Die für dieses Szenario gewählten Typen richten sich nach den Daten der Verkehrszählung. Um möglichst homogene Daten zu haben, wurden im weiteren Verlauf die Daten aus der HBEFA 3.3 Schnittstelle verwendet.

2 Erstellung des Szenarios in SUMO

Die Open-Source Software "Simulation of Urban MObility" (SUMO) erlaubt die kontinuierliche Verkehrssimulation in großen Graphen-Netzen [10].

2.1 SUMO OSM Karte

Es wurde mittels des osmWebWizards ein Szenario erstellt. Dieses Szenario umfasst den Großteil des Stadtgebietes von Berlin.



Abbildung 1: Auswahl des Simulationsbereichs in osm-WebWizard

Die Einstellungen für das Szenario wurden wie folgt ausgewählt:

	Through Traffic Factor	Count
Cars	5	12
Trucks	5	8
Bus	5	4
Motorcycles	2	4
Trams	20	2
Urban Trains	40	2

Tabelle 1: Parameter der Simulation

Der *Through Traffic Factor* legt fest, wie oft es wahrscheinlicher ist, dass eine Kante am Rand des Simulationsgebiets gewählt wird als eine Kante, die sich vollständig innerhalb des Simulationsgebiets befindet. Der *Count* Parameter definiert, wieviele Fahrzeuge pro Stunde / Spuren Kilometer erzeugt werden. Würde unser Beispiel aus drei Kanten mit einer Länge von fünf Kilometern bestehen die jeweils zwei Spuren haben und der Count würde auf 90 gesetzt werden ergibt sich daraus: $5 \cdot 2 \cdot 90$

= 900 Fahrzeuge die pro Stunde generiert werden würden. Über den Parameter *randomTrips* lässt sich die Zeit einstellen, wie oft ein neues Fahrzeug im Netzwerk generiert wird. In dieser Simulation wurde mit einem *randomTrips* Parameter von 1 gearbeitet. [10] Als Simulationsoptionen wurden Import Public Transport, Car-only Network und Add Polygons (1000 Duration) ausgewählt. Als Vereinfachung wurden die Wasserwege sowie Fußgänger und Fahrradfahrer aus der Simulation entfernt.

2.2 SUMO Konfiguration

Die Fahrzeugkonfigurationen mussten für das Emissions-Modell erweitert werden. Dazu werden die *vehicle-Types* mit den jeweiligen Emissions-Klassen nach HBEFA 3 gekennzeichnet [10]. Die Emissions-Klassen für die jeweiligen Typen wurden wie in Tabelle 2 ausgewählt.

Klassen nach HBEFA	
Cars	HBEFA3/PC_G_EU6
Trucks	HBEFA3/HDV
Bus	HBEFA3/BUS
Motorcycles	HBEFA3/PC_G_EU5

Tabelle 2: gewählte HBEFA Emissions-Klassen

Bei den Motorrädern fiel die Entscheidung für alle Typen auf die Emissionsklasse EU5, da seit 2020 alle neuen Typenzulassungen diese Norm erfüllen müssen. Für Trams und S-Bahnen wurden keine Emissionsklassen gewählt, da diese im Modell zwar als Verkehrsteilnehmer simuliert werden, jedoch die Emissionen sich auf den Verkehr von Kraftfahrzeugen beschränken soll und daher die Emissionen der Züge vernachlässigt wird.

Damit die Kanten- bzw. Spurenemissionsdaten im Detail simuliert werden, wurde eine zusätzliche Konfigurationsdatei angelegt (siehe Abbildung 2). Diese beinhaltet die Konfigurationen in welchen Dateien die jeweiligen Emissionsdaten abgelegt werden sollen.

```

1 <additional>
2   <edgeData id="1" file="edge.traffic.dump.xml"/>
3   <edgeData id="edgebased" type="emissions" file="edge.emission.dump.xml" excludeEmpty="true"/>
4   <laneData id="lanebased" type="emissions" file="lane.emission.dump.xml" excludeEmpty="true"/>
5 </additional>

```

Abbildung 2: Zusätzliche Konfigurationen in *additional.xml*

Die Simulation kann dann durch Aufruf des SUMO-Simulators gestartet werden. Um die Rechenzeit zu begrenzen wurde die Simulationsdauer für die hier vorgestellten Simulationen auf 100 Simulationsschritte festgelegt. Dabei entspricht ein Simulationsschritt einer Sekunde da der Parameter *-step-length* nicht verändert wurde. Um genauere Ergebnisse zu erzielen wird zur späteren Validierung die *step-length* auf 0.01 gesetzt was 10ms entspricht. Mit einer *step-length* von 1 werden hinreichend viele Datensätze generiert, um die statistischen Mittelwertbildungen zu rechtfertigen.

```

sumo -c osm.sumocfg --emission-output
osm.emissions.xml --end 100 --additional-
files additional.xml

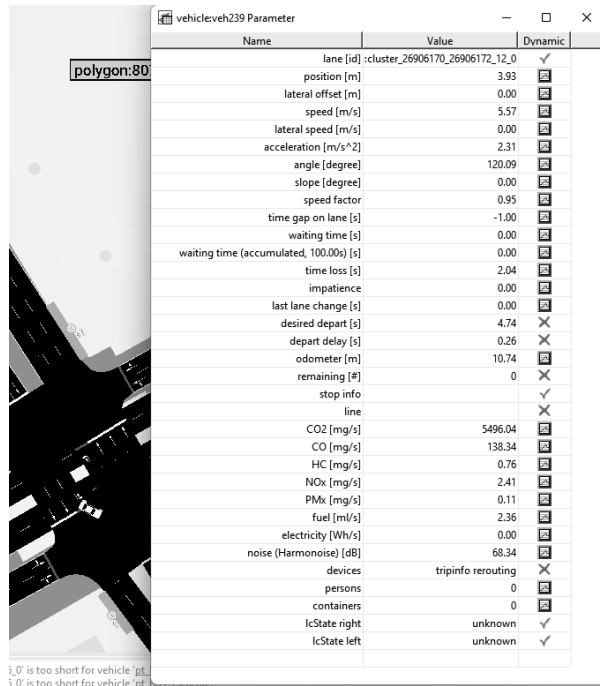
```

Während der Simulation werden Fahrzeuge bei jedem weiteren Simulationsschritt hinzugefügt und auf eine zufällige Route geschickt, dadurch entsteht mit der Zeit eine homogene Verteilung der Fahrzeuge im Straßennetz. In einem weiteren Schritt wurden Fahrzeuge in die laufende Simulation mittels TraCI geladen, die einer festen Route folgen. Dadurch soll es ermöglicht werden, die Daten aus der Verkehrszählung nachzubilden, indem der DTV_{WT} KFZ auf den größeren Hauptverkehrsstraßen möglichst genau nachgebildet wird. Die Validierung dieser Ergebnisse dauert jedoch noch an, da für möglichst realitätsnahe Ergebnisse die Simulationsschritte erhöht werden müssen.

3 Simulierte Emissionsdaten

Nachdem die Simulation durchlaufen ist, sind die Emissions sowie die Verkehrsdaten in drei XML-Dateien einsehbar. Bei 100 Simulationsschritten werden ca 10000 Fahrzeuge simuliert. Dabei wurden ca. 10000 Routings mittels Dijkstra über 46400 Kanten durchgeführt. Die Dauer der Simulation betrug zwischen 800 und 900s.

In *lane.emission.dump.xml* sind die kumulierten Emissionsdaten für die einzelnen Fahrstreifen einsehbar. Die vom Fahrzeug zurückgelegte Strecke lässt sich in einer separaten XML-Datei nachvollziehen für jeden einzelnen Simulationsschritt.



Name	Value	Dynamic
lane [id]	cluster_26906170_26906172_12_0	☒
position [m]	3.93	☒
lateral offset [m]	0.00	☒
speed [m/s]	5.57	☒
lateral speed [m/s]	0.00	☒
acceleration [m/s^2]	2.31	☒
angle [degree]	120.09	☒
slope [degree]	0.00	☒
speed factor	0.95	☒
time gap on lane [s]	-1.00	☒
waiting time [s]	0.00	☒
waiting time (accumulated, 100.00s) [s]	0.00	☒
time loss [s]	2.04	☒
impatience	0.00	☒
last lane change [s]	0.00	☒
desired depart [s]	4.74	☒
depart delay [s]	0.26	☒
odometer [m]	10.74	☒
remaining [#]	0	☒
stop info		☒
line		☒
CO2 [mg/s]	5496.04	☒
CO [mg/s]	138.34	☒
HC [mg/s]	0.76	☒
NOx [mg/s]	2.41	☒
PMx [mg/s]	0.11	☒
fuel [ml/s]	2.36	☒
electricity [Wh/s]	0.00	☒
noise (Harmonoise) [dB]	68.34	☒
devices	tripinfo rerouting	☒
persons	0	☒
containers	0	☒
lcState right	unknown	☒
lcState left	unknown	☒

Abbildung 3: simulierte Parameter eines PKWs in SUMO

Durch die gewählten Parameter beim Starten der Simulation wurden die Emissionsdaten für Kanten und Spuren in unterschiedliche Dateien abgelegt. Damit stehen jetzt zusätzlich zur eigentlichen Verkehrssimulation auch die Emissionsdaten zur Verfügung. Die Daten sind detailliert abrufbar für die einzelnen Kanten des in SUMO simulierten Verkehrsnetzes und lässt daher eine Auflösung bis auf die jeweilige Fahrspur zu (siehe Abbildung 4). In Abbildung 4 werden die simulierten Emissionswerte für eine Fahrspur sichtbar.

```
<lane id="-1002878124_0" sampledSeconds="2.26"
CO_abs="37.028007" CO2_abs="23366.079058"
HC_abs="0" PMx_abs="4.840656" NOx_abs="220.647828"
fuel_abs="9.906021" electricity_abs="0"
CO_normed="13330.082539" CO2_normed="8411788.460820"
HC_normed="0" PMx_normed="1742.636300" NOx_normed="79433.217922"
fuel_normed="3566.167529" electricity_normed="0"
traveltime="0.03" CO_perVeh="0.514278" CO2_perVeh="324.528876"
HC_perVeh="0" PMx_perVeh="0.067231" NOx_perVeh="3.064553"
fuel_perVeh="0.137584" electricity_perVeh="0"/>
```

Abbildung 4: Auszug simulierte Emissionsdaten in lane.emission.dump.xml

Die simulierten Emissionsdaten eines beliebigen Fahrzeugs können während der laufenden Simulation abgelesen werden (Abbildung 3). Auf der Basis dieser netzbezogenen Auswertung kann dann den einzelnen Fahrzeugen ein individueller Emissionswert für den der Kante entsprechenden Routenabschnitt zugewiesen werden.

Aufsummiert über alle Kanten einer Route errechnet sich daraus schließlich die Gesamt-Emission einer Fahrt.

4 Diskussion

In diesem Projekt konnte ein Verkehrsmodell von Berlin als Szenario erstellt werden, welches die Emissionsdaten nach HBEFA 3.3 Standard simuliert und für weitere Arbeitsschritte verfügbar macht. Die simulierten Emissionsdaten können mit Hilfe der sumo-gui visualisiert werden, sodass Straßen mit hohen Emissionen hervorgehoben werden. Die Projektphasen eins und zwei können als erfolgreich abgeschlossen betrachtet werden. Damit ist die Machbarkeit für diese beiden Phasen gezeigt. Jedoch sind die 100 Simulationsschritte derzeit noch nicht repräsentativ für das komplette Verkehrsmodell. Die Anzahl der Simulationsschritte muss im späteren Verlauf erhöht werden, um die tatsächlichen Auslastungsdaten und Fahrzeiten möglichst realitätsnah nachzubilden. Phase Drei, also der Abgleich der simulierten Daten mit den bei der Verkehrszählung tatsächlich gemessenen, befindet sich derzeit in der Validierung.

Mit Abschluss von Phase Drei liegen dann eine vollständige Parametrisierung des Netzes und des Lastmodells für Berlin und damit validierte, Kanten- und Fahrspur-spezifische Daten eines Normal-Verkehrs vor. Diese Arbeiten sollen sich zunächst räumlich begrenzt für einen ausgewählten Berliner Bezirk durchgeführt werden, um Rechenzeit sowie Validierungszeit zu begrenzen. Die randomisierten Fahrten sollen die Fahrten der Grundlast im Verkehr abbilden. Problem dabei ist, dass diese Fahrten zwar durch die Verkehrszählung abgebildet werden, jedoch beschränken sich diese simulierten Verkehre des DTV_{WT} KFZ nur auf die Hauptverkehrsstraßen. Um auch eine mögliche Verkehrslast in Nebenstraßen zu erzeugen, wurden zusätzlich zufällig fahrende Fahrzeuge implementiert. Die Effekte dieser zusätzlich generierten Fahrzeuge die während des Simulationslaufes dynamisch und zufällig mittels TraCI hinzugefügt werden, um den Faktor DTV_{WT} KFZ nachzubilden, konnten allerdings noch nicht vollständig validiert werden. Erst wenn auch dieser Schritt abgeschlossen sein wird, ist eine dynamische Last im Verkehrsnetz hinreichend valide nachgebildet, auf deren Basis dann das eigentlich angestrebte, emissionsbasierte Routing für das ganze Berliner Model durchgeführt werden kann. Dabei werden weitere Annahmen über die unterschiedlichen Lastprofile im Tages-/Wochen-/Jahres-Zyklus zu berücksichtigen sein.

5 Fazit und Ausblick

Inhaltlich erscheint das Projekt inzwischen soweit fortgeschritten, dass eine erfolgreiche Implementierung auch des fehlenden Schrittes Vier, also der eigentlichen Routing-Aufgabe unter Berücksichtigung der Emissionen eines individuellen Fahrzeugs bei der aktuellen Verkehrslage im Netz, absehbar erscheint.

Bei den bisherigen Experimenten haben sich allerdings die erheblichen Rechenzeiten, die im Wesentlichen durch die Implementation mittels TraCI begründet sind, als sehr hinderlich für einen praktischen Einsatz erwiesen. Daher wird aktuell darüber nachgedacht, zur Effizienzverbesserung bezüglich der Simulationszeiten das Vehicles in Network Simulation (VeinS) Framework [8] zu implementieren.

References

- [1] Umweltbundesamt, *Nationale Inventarberichte zum Deutschen Treibhausgasinventar 1990 bis 2016 wie Nahrzeitprognose für 2017*.
- [2] Bundesministerium für Verkehr und digitale Infrastruktur: *Verkehr in Zahlen 2017/2018*, Hamburg, ISBN 978-3-87154-617-4
- [3] Umweltbundesamt: Emissionen des Verkehrs: <https://www.umweltbundesamt.de/daten/verkehr/emissionen-des-verkehrs> abgerufen am 19.05.2022
- [4] Umweltbundesamt: *Transport Emission Model, Daten- und Rechenmodel TREMOD 5.81*.
- [5] Helmers, E. (2010). *Bewertung der Umwelteffizienz moderner Autoantriebe – auf dem Weg vom Diesel-Pkw-Boom zu Elektroautos*. Umweltwissenschaften und Schadstoff-Forschung, S. 564–578.
- [6] Bundesregierung. (01. Oktober 2012). Bundesregierung: *Pressekonferenz nach dem Spitzengespräch zur Elektromobilität*.
- [7] Engelmann M., Schulze P., Wittmann J., *Emissionsbasierte Routenführung mit Hilfe der GraphHopper API und OSM*, 2019, DOI: 10.1007/978-3-030-30862-9_7
- [8] Verkehrsmengen DTV 2014: <https://www.berlin.de/umweltatlas/verkehr-laerm/verkehrsmengen/2014/zusammenfassung/> (Abgerufen am 20.05.2022)
- [9] VeinS Framework: <http://veins.car2x.org> (Abgerufen am 22.05.2022)
- [10] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner, *Microscopic Traffic Simulation using SUMO*, IEEE Intelligent Transportation Systems Conference (ITSC), 2018
- [11] Volker Wohlgemuth: *Machine learning for sustainable logistics*, Environmental Informatics: Techniques and Trends Adjunct Proceedings of the 32nd edition of the EnviroInfo - the long standing and established international and interdisciplinary conference series on leading environmental information and communication technologies, S.250 ff., Shaker Verlag, Herzogenrath, 2018, ISBN: 978-3-8440-6138-3

Diskrete Systeme

Discrete Systems

A Simulator for NSA-DEVS in Matlab (L)

D. Jammer, P. Junglas, T. Pawletta, S. Pawletta

Adapting to Change of Model Transitions in Proxel Based Simulation of CHnMMs (L)

D. Bodnár, C. Krull

A Hybrid User Model for Virtual Stochastic Sensors (L)

C. Krull

Analysis of Train Delays using Bayesian Networks (K)

V. Barzal, M. Rößler, M. Wastian, F. Breitenecker, N. Popper

Parameter-Free Approximation Method for Controlling Discrete Event Simulation by Reinforcement Learning (K)

D. Pasterk, A. Körner

Die Langbeiträge (L) sind zu finden im **ARGESIM Report 20, Proceedings Langbeiträge ASIM SST 2022**

ISBN ebook 978-3-901608-97-1, DOI: 10.11128/arep.20, ASIM Mitteilung AM 180

Analysis of Train Delays using Bayesian Networks

Valentin Barzal^{1*}, Matthias Rößler², Matthias Wastian²,

Felix Breitenecker¹, Nikolas Popper³

¹Institute of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria; *valentin.barzal@tuwien.ac.at

²dwh GmbH, Neustiftgasse 57-59, 1070 Vienna, Austria;

³Institute of Information Systems Engineering, TU Wien, Favoritenstraße 11, 1040 Vienna, Austria;

Abstract. Bayesian networks can be used for analysis and representation of dependencies in large data sets. Due to their property of operating with graphs, they are suitable for analyzing delays in rail networks. After getting an overview of the theory of Bayesian networks, this article deals with recent literature about Bayesian networks and train delays. Furthermore, the presented methods will be applied to data from the Austrian railway network.

Introduction

Since public transport plays a major role in society, it is on societies behalf to keep delays to a minimum. In order to reach the goal of minimizing delays, it is important to understand not only the reasons of delays but also the propagation of delays within the public transport network. The availability of large data sets, powerful computers and suitable mathematical tools allows analyzing such delays in a sensible way.

1 Bayesian Network

A *Bayesian network* is a model represented by a directed acyclic graph (DAG) $\mathcal{G} = (V, E)$, where V is the vertex set and E is the edge set. Every vertex represents a random variable. There exists an edge $e \in E$ between two vertices $v, w \in V$, if and only if there is a probabilistic dependency between the two random variables represented by v and w ([1]).

Depending on the model, the random variable (i.e. vertex) can take on different types of values ([2], [3]), e.g. unordered values (e.g. *blue*, *green*, *yellow*), ordered values (e.g. *low*, *medium*, *high*) or continuous values ($\mathbb{R}$). Every random variable also has its probability distribution, which needs to be added to the graphical rep-

resentation. If the random variables take on continuous values (e.g. when dealing with continuous data), we often consider *multivariate normal variables*, i.e. the random variables are normally distributed and linked by linear constraints. An example of this case is illustrated in Figure 1.

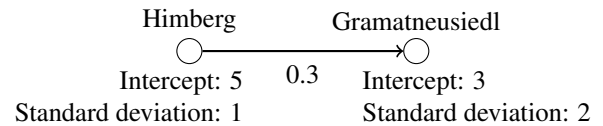


Figure 1: Considering that a train at station *Himberg* is 10 minutes late, then it is expected to be 6 minutes late ($3 + 0.3 \cdot 10 = 6$) at station *Gramatneusiedl*, with a standard deviation of 2 minutes.

When using Bayesian networks for modelling real-world problems, we often just have data describing the values of our random variables, but no information about the graph structure, i.e. the edges/dependencies between the vertices/random variables. In this case, we want to derive the graph structure from the values of the random variables. The *Max-Min Hill-Climbing* method (MMHC) enables this derivation and will be used in Section 2 and Section 3. A detailed introduction to this algorithm can be found in [4].

2 Analyzing the Long Island Rail Road System

In [1], Ulak et al. investigate network-wide pattern of rail transit delays in the Long Island Rail Road System, i.e. a railway system in New York that has a length of 620 miles and consists of a total of 11 rail lines. In

Section 3, the methods used in [1] will be modified and used for data of the Austrian railway system. Therefore, Section 2 recapitulates the methods and ideas of [1] without going too deep into detail or presenting the results.

In order to build a Bayesian Network as described in Section 1, every station is considered as a random variable. It takes on the average delay at this station within a certain time interval. There should exist an edge between two vertices (i.e. between two stations) if and only if there is a dependence between those two vertices (i.e. if a delay at one station affects the occurrence of a delay at the other station).

It is possible to classify each train as either eastbound or westbound. When analyzing the delay propagation, i.e. when creating the Bayesian Network, either all eastbound or all westbound trains are considered, but not both east- and westbound trains.

By analyzing the data of an app used by the passengers, the authors receive the delay of every train at every station (with a few exceptions). The observation period is divided into 1-hour intervals. For every station the average delay per train within the examined hour is calculated. A data sample consists of the average delay time at every station within one hour. By using the *Max-Min Hill-Climbing* method (MMHC) and the maximum likelihood method, the graph structure and the corresponding parameters are compiled from the independent data samples. The R-package `bnlearn` was used for this task. Detailed documentation about this package can be found in [3].

In order to receive an expressive score for every stations role in delay propagation, two metrics are introduced. The Inducer score of station s indicates how much a delay at station s influences the delays at all the other stations. The Susceptible score of station s indicates how much the delay at station s is influenced by the delays at all the other stations. Similar scores will be introduced in Section 3 in a more precise way.

3 Analyzing the Austrian Railway System

In this section we want to modify the methods of [1] and apply them to data of the Austrian railway network. For this purpose, delay data provided by the Austrian national railway company *Österreichische Bundesbahnen* (ÖBB) is used. It not only contains information about the delay at every train station, but at every operation

control point, a so-called "Betriebsstelle". These are specific points within the network. Every train station is also a operation control point. In the following, the word "station" will be used for operation control point.

3.1 Differences to the Long Island Rail Road System

The different topology of the railway network and the different timetables cause different problems that need to be faced:

- In [1], the length of every time interval is chosen to be 1 hour. Considering longer time intervals "flattens" the delays, since temporal peaks are divided by a higher number of trains. So shorter time intervals provide better "resolution" of the occurrence of delays.

On the other hand, it takes some time to propagate the delay within the network. In Austria, there are longer distances between stops and therefore it takes longer to propagate delays. Choosing the length of the time interval to be 1 hour, most trains will "occur" in more than one time interval. Since the different data sets (i.e. delays within different time intervals) are meant to be independent, this causes some problems.

- There are stations that are not passed by any train in certain time intervals. Calculating the average delay in that period would provoke dividing by zero. This problem can be solved by considering the totaled delay instead of the average delay.

However, this still might yield meaningful results, since the delays at a station that is approached by a lot of trains, each having just a little delay, might be more relevant than the delay at a station that is only approached by a single train, having a longer delay.

- The Long Island Rail Road System is quite crowded, i.e. if a train is delayed, it is very likely that this affects other trains. In Austria, there are also parts where the temporal distance between different trains is much longer. In this case, delays will still propagate since a delayed train probably will still be delayed at its next stop, but it does not directly affect the delays of other trains.
- By considering only the eastbound or westbound trains, a cycle-free network was obtained in [1]. In

Austria, a classification in eastbound or westbound is not possible (e.g. there are also trains going from north to south and vice versa). However, considering all the trains will lead to cycles within the network. This conflicts with the attempt of modelling the network as a directed *acyclic* graph. A strategy to resolve this problem is to delimit the number of trains considered.

- In the Long Island Rail Road System, there are just 124 stops, so it is a relatively small network. Considering all trains in Austria leads to a gigantic model (in terms of vertex numbers, edge numbers and parameters). But when examining only a selection of trains (and its corresponding stations), there are still influences from the non-considered trains. So there is "external" disturbing.
- There are also trains not operated by ÖBB using the same tracks. In the provided data there is only information about the trains operated by ÖBB. So even if no data selection is done, there will be still disturbances by other, not observed trains.

3.2 Methodology

Considering Subsection 3.1, the following restriction will be made:

- Only trains going from Vienna (station *Hauptbahnhof* or *Westbahnhof*) to St. Pölten and vice versa and the corresponding operational points between those two stations will be observed. In this case each train can be classified in eastbound or westbound and we obtain cycle-free networks. This section is also quite crowded, so there will be interactions between different trains.
- Since the distance between Vienna and St. Pölten is relatively short, we can set the length of the time intervals to 1 hour, without dealing too much with the problems mentioned in Subsection 3.1.
- We will not consider the average delay but the totaled delay of every time interval at each station.

Considering only the stations from Vienna to St. Pölten and vice versa is of course a very strong restriction but it enables us receiving suitable results. The considered stations ("Betriebsstellen") are sketched in Figure 2. In Vienna, there are two branches - one branch going to *Wien Westbahnhof* and one branch going to *Wien*

Hauptbahnhof. Between the stations *Knoten Hadersdorf* and *Knoten Wagram* there are also two different routes: In the north (via *Tullnerfeld*) there is the so-called *Neue Westbahn*, in the south (via *Pressbaum*) there is the *Alte Westbahn*.

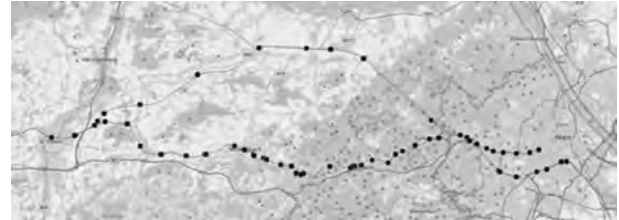


Figure 2: Considered stations (operation control points) between Vienna and St. Pölten

3.3 Network structure

Similar to [1], we use the R-package *bnlearn* (and the functions *hc* and *bn.fit*) to obtain the network structure. It was also considered to use the Python-packages *Pomegranate* and *PyMC3* for the same task, but those packages can only deal with random variables that take on unordered values. The results can be seen in Figure 3 and Figure 4. In these figures, the directions of the dependencies were not indicated.

When searching for the graph structure, we expect a graph that resembles the geographical topology of the stations, e.g. it would not make sense if a station close to St. Pölten directly affects a station close to Vienna. The topological sorting never coincides with the actual geographical order of the stations. However, we did expect such a behavior, since phenomena like backward propagation (i.e. dependencies going in the opposite direction) and others can cause a mismatch with the geographical order of the stations. Furthermore, some stations are very close and therefore have almost the same delays. In this case, the order of those stations in the obtained network is almost arbitrary.

When observing the plots visually, we detect that there are also connections between stations that are far apart. There are even connections between stations from the *Alte Westbahn* and *Neue Westbahn*, whereat these kind of connections are relatively rare.

3.4 Metrics

The metrics are inspired by the metrics used in [1], but slightly modified.

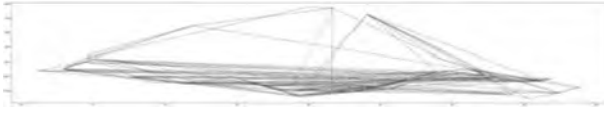


Figure 3: Network connections (eastbound). X-axis and y-axis indicate latitude and longitude.

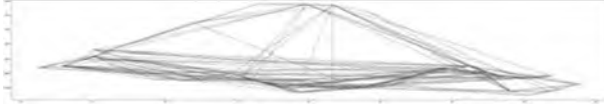


Figure 4: Network connections (westbound). X-axis and y-axis indicate latitude and longitude.

The Inducer score indicates how much a delay at a station s influences the delays at all the other stations. Therefore, the delay at the examined station s is set to the median delay of this station (based on the data) and the delays at all the other stations are set to 0. Given this initial condition, the expected delay at every other station is estimated by the Bayesian network. Those values are averaged over all stations. The score of station s can be written as

$$\text{Score}_I(s) = \frac{1}{n-1} \sum_{\substack{i=1 \\ i \neq s}}^n \mathbb{E}(X_i | X_s = x_s, \mathbf{X} \setminus (X_s \cup X_i) = 0)$$

where n is the number of random variables, x_s is the median (observed) delay at station s and $\mathbf{X} = (X_i)_{i \in \{1, \dots, n\}}$ is the ensemble of all random variables.

The Susceptible score indicates how much the delay at a station s is influenced by the delays at all the other stations. Therefore, the delay at one station i (with $i \neq s$) is set to its median delay (based on the data) and the delays at all the other stations (including the examined station s) are set to 0. Given this initial condition, the expected delay at the examined station is estimated by the Bayesian network. This procedure is repeated and averaged over all stations $i \neq s$. Using the same notation as above, the score of station s can be written as

$$\text{Score}_S(s) = \frac{1}{n-1} \sum_{\substack{i=1 \\ i \neq s}}^n \mathbb{E}(X_s | X_i = x_i, \mathbf{X} \setminus (X_i \cup X_s) = 0)$$

When considering the metrics, we observe that almost all stations with the highest (top 4) scores are either stations that are passed by all trains or stations of

the *Alte Westbahn*. Only the station *Tfo* (close to Tullnerfeld) has the highest inducer score (westbound), although it is located at the *Neue Westbahn*. So the *Neue Westbahn* acts well in reducing delay propagation. Going in the westbound direction, two stations at the very beginning of the route reach high susceptible scores (*Wien Hauptbahnhof* and *Wien Westbahnhof Frachtenbahnhof*). Those stations are very susceptible to delays at other stations (although almost all stations are passed *after* those two stations). We also observe that some stations reach negative susceptible scores. According to our model, this means that delays at certain other stations reduce the delay at this observed station.

3.5 Conclusion

When applying the methods introduced in [1] to the Austrian railway system, we need to make restrictions. In order to obtain the inevitable property of an acyclic directed graph, we can only consider an area where every train can be classified into a certain (cycle-free) direction. R yields a network structure that seems partly reasonable. There are also connections between stations that are far apart and connections between *Alte Westbahn* and *Neue Westbahn*. However, this does not necessarily need to conflict with the real behaviour.

The inducer score and susceptible score indicate which stations play a big role in delay propagation. It seems reasonable that the *Neue Westbahn* acts well w.r.t. reducing delay propagation, since this section was constructed quite recently. The fact that stations at the beginning of the route have quite high susceptible scores fits to the idea of back-propagation. However, there are also stations with negative susceptible scores. This does not seem to be reasonable.

References

- [1] Ulak MB, Yazici A, Zhang Y. Analyzing network-wide patterns of rail transit delays using Bayesian network learning. *Transportation Research Part C: Emerging Technologies*. 2020;119:102749.
- [2] Korb KB, Nicholson AE. *Bayesian artificial intelligence*. CRC press. 2010.
- [3] Scutari M, Lebre S. *Bayesian networks in R: with applications in systems biology*. 2013.
- [4] Tsamardinos I, Brown LE, Aliferis CF. The max-min hill-climbing Bayesian network structure learning algorithm. *Machine learning*. 2006;65(1):31–78.

Parameter-Free Approximation Method for Controlling Discrete Event Simulation by Reinforcement Learning

Daniel Pasterk¹, Andreas Körner¹

¹Institute of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria

Abstract. A novel k NN-based approximation method for Reinforcement Learning (RL) is used to control and optimize a Discrete Event Simulation (DES). The method does not require design parameters, is suitable for unknown and new simulation environments, and can handle irregular and partially sparse state space. We show in a demonstrative queuing simulation that the method is more robust than artificial neural networks and achieves comparable performance.

1 Introduction

In this contribution we want to show how Discrete Event Simulations can be controlled and optimized using reinforcement learning. From a general perspective, it is promising to optimize simulations using RL. It provides the ideal framework to represent time-dependent decision problems in a very natural approach, while the simulations can generate sufficient data. Changes in the dynamics of environment can also be addressed via different exploration strategies. Although a large number of powerful algorithms have emerged since the 1960s, the broad availability of powerful computational resources, has extended its applicability in many areas. Starting with the impressive achievements of DeepQ-Networks [1] in 2013, this form of machine learning was becoming increasingly popular. Controlling a simulation with RL is follows the framework of Markov Decision Processes (MDPs), where an agent interacts with an environment. The agent receives the state of the environment and decides on an action, while the environment responds with a new state and a reward. The agent tries to choose his actions in a way that the cumulative reward is maximized. The simulation therefore represents the environment, in which the agent interacts within the described loop.

While the basic architecture is very clear, the dif-

ficulty often arises at dealing with the formal requirements of the framework. It is a important requirement to represent the simulation as an MDP. While the optimization goals and control options can often be represented very directly with a corresponding reward and action set, the so-called Markov property must be fulfilled for the state variable: The entire (relevant) information must be able to be encoded in a single state which has to be independent of its history. In order to fulfill these preconditions feasibly, there are various approaches: The simulation can already be developed with the background of this requirement and the corresponding state variables can be equipped with a well-suited encoding. Though it has to be mentioned here that special knowledge about the nature of the simulation is relevant. This somewhat undermines the claim of a model-free method. As an alternative to the "hand-crafted" states, one can take the internal simulation variables directly. These can basically be used as states in the MDP framework and should satisfy the Markov property, but it sadly leads to a very high complexity. In addition, the resulting state space is very irregular - many combinations of the individual components of the state vector are not even possible due to the simulation logic. Large areas of the state space are therefore not used, while in other areas there is a lot of information in a very dense form. Hence, a suitable approximation method has to be found, which can handle these specific properties sufficiently well. This contribution presents a non-parametric approximation method based on the idea of k -nearest-neighbor classification algorithm, which is very well suited for the mentioned challenges. We demonstrate the method in the context of classical Q-learning and Monte-Carlo algorithms.

2 Related Work

We want to give a small selection of publications, which illustrate the successful application and then go into specific limitations: In 1998, Mahadevan and Theodorou [2] use the customized algorithm "SMART" to optimize transfer lines from a fab including predictive maintenance. The results show superior performance compared to the classical 'KANBAN' heuristic. In Waschneck et al. (2018) [3], DQN is successfully applied to control a digital twin of a semiconductor fab. The DQN actions control specific dispatch heuristics to increase overall performance. Shuhui Qu, Jie Wang, and Shivani (2016) [4] use RL to solve a dispatching problem in factories. A simulation was created special as MDP and the learned rules are more cost effective than most well-known heuristics. Doltsinis, Ferreira, and Lohse (2014) [5] use RL to optimize a production startup in an 'automatic assembly station'. The use of a batched Q-learning algorithm shows the general suitability of this decision-supported approach. Some selection of general approaches, which focuses on the general connectivity of RL with simulations: Capocchi et al. (2022) [6] show how DEVS properties can be used within a RL integration focusing on temporal, hierarchical and multi-agent aspects. Lang et al. (2021) [7] demonstrate the application of DEVS to a production planning problem. In this approach, it is shown how control can be performed using a standardised interface (OpenAi Gym). In Greasley (2020) [8], various software products from simulation and machine learning are examined for connectivity. Some interesting considerations are also made about the structure of a useful connectivity component. In Choo et al. (2020)[9], the general problem that discrete event simulations cannot be directly translated into MDP is pointed out. It is proposed to translate the state space and action space of the simulation into their counterparts in the RL formalism using equivalence classes. The approach sounds interesting, but unfortunately no statement is made about the concrete form of the relationship.

3 Method

3.1 Architecture

The architecture of our setup can be summarized in Fig.1. For the interface of the simulation we use the interface of OpenAi-Gym [10], by which the states, ac-

tions and the reward are exchanged. The simulation-internal variables are made accessible to the agent as a state vector without further processing. The agent has no further information about the model apart from the state vector and the simulation is technically controlled by this model-free agent. Both the implementation of the code for the approximation and learning feature, as well as the implementation of the simulation was done in the programming language Julia. The framework "SimJulia" was used for the simulation and the modeling follows the agent-based paradigm.

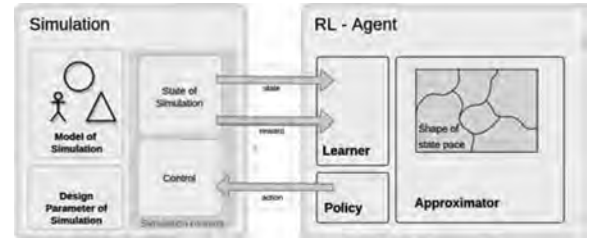


Figure 1: General overview of the architecture of the simulation and the RL agent. The approximator must represent the underlying state space sufficiently well.

3.2 Approximation Method for Reinforcement Learning

On the agent side, we strongly separate between the elements of "learner", "approximator" and "policy". This is also illustrated in Fig.1. For the case-study below, we use the update rules from classical Q-learning. For the state space S , action space A , $\alpha \in \mathbb{R}$, $\gamma \in \mathbb{R}$ and a possible reward $r \in \mathbb{R}$ the update rule can be written as

$$q(s, a) := q(s, a) + \alpha(r + \gamma \max_{a' \in A(s')} (q(s', a')) - q(s, a)). \quad (1)$$

with $s \in S$ and $a \in A$. We use this update rule in connection with various approximation methods such as "State Aggregation", "Tile-Coding" and "Linear Regression".

We also use Monte-Carlo-1st-Visit [11] and a Deep Q-Network [1] with two different capacities as a reference.

Since A is finite, an optimal policy π is immediately calculated as

$$\pi(s) := \operatorname{argmax}_{a \in A(s)} q(s, a), \quad (2)$$

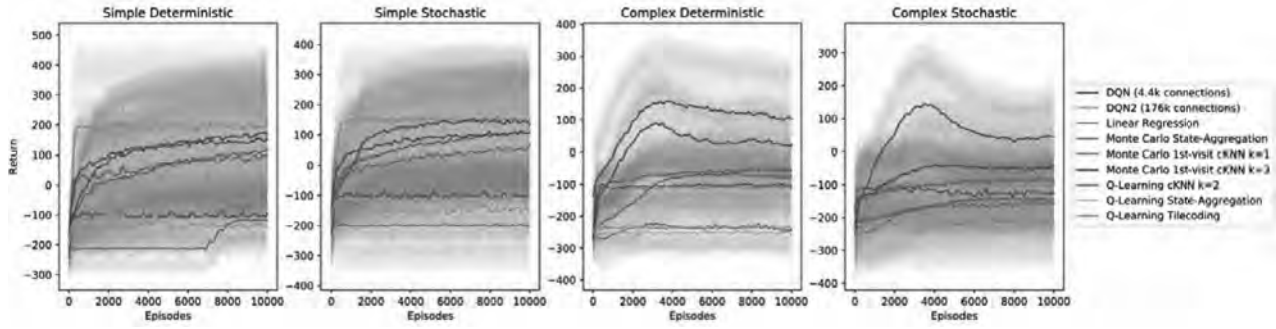


Figure 2: Performance of k -NN approximation in combination with Q -Learning and first-visit Monte Carlo compared to other common approximation methods, in simple-deterministic ($n = 20$), simple stochastic ($n = 20$), complex-deterministic ($n = 20$), and complex-stochastic tasks ($n = 20$).

where ties are broken randomly, after having computed the action-value function q .

Our method for approximating $q(s, a)$ is based on the idea that the collected interactions between agents and environment are grouped into $(k-d)$ -Trees to the respective action $a \in A$. The RL update rules provide an error that is added to the current estimate of an existing query. To obtain $q(s, a)$ in a query, every KD-tree is searched individually for the k nearest neighbors of a state for each action and the q -value is calculated by the arithmetic mean over the k stored values.

4 Queuing Simulation as Case Study

4.1 Simulation Description

We consider 4 different scenarios from a typical queuing simulation at a service desk. In the simple model, the agent can change the configuration once in each time window between 0 and 5 open cash registers, while in the more complex model 10 changes can be made per time window and it is possible to open up to 7 cash registers simultaneously. The service time for each customer is 0.25 hours, and the staff budget is 5 hours in both cases. In the stochastic case, the arrival times are sampled from a Poisson distribution where we have peaks in the later morning and in afternoon. The patience p of the customers is sampled from a uniform distribution between 3 and 5, and the service time is sampled from a normal distribution with mean 0.25 hours and standard deviation $\sigma = 0.05$ hours. In the more complex case, the agent can make a decision and rede-

ploy staff 10 times per hour. Therefore, the complexity increases slightly. This simulation is designed in such a way that in the deterministic case, 60 customers visit the bank per day, and 15 staff hours would theoretically be required to serve for all customers. However, there are far too few resources. The state-space of the agent consists of the time of day, the length of the queue, and the number of open desks. One episode is divided into ten-time units. A reward of +10 is received for a completed request and -10 for an abort.

4.2 Experimental Results

Over 10k episodes were simulated for all 4 scenarios. Mean and standard deviation were determined over 20 independent runs. Known approximation methods from the literature were taken up as a comparison. These were configured to the best of our knowledge. The results can be seen in Fig.2.

In the simple case, classical DQN can adapt very well to both the deterministic and the stochastic variants. Linear regression, even in the simple case, is able to provide useful state approximation only at a very late stage. In the more complex case, the kNN variants lead very quickly to a good policy, while other methods would probably require further and more costly adaptation of the design parameters. We can verify that the approximation by the k NN method works sufficiently well in both cases and that classical methods by state aggregation do not have sufficiently well approximation properties in connection with RL. Neural networks also provide a good performance, but were complex to configure in order to achieve a learning behavior.

5 Discussion and Conclusion

We observe that the state space approximation by k -NN is superior to classical simple methods. In performance comparison with DQN, the universal approximation property of neural networks can provide good performance here. A fairly big advantage of this method in the context of simulations is the robustness against proper ranges of values in the state representation and the necessary design parameters in the approximation itself. These are serious advantages in the development and actual usage as an optimization method. While in parametric approximation methods one has to find a very balanced dimensioning for a good performance, this is not necessary in the present case. Furthermore, the simulation variables can be taken directly and do not need to be specially prepared by some pre-processing. This is remarkable especially in the case of neural networks, where the performance can be very relevant mostly due to the size of the network and also the encoding of the information in the first layer. While the k NN approximation does not require any design parameters, for the other methods a complex evaluation process was necessary to show any learning behavior at all. Especially the choice of the neural network leaves only a narrow range between over and underfitting. Therefore, we see the advantages of the k -NN approximation in connection with classical RL algorithms especially in the parameter-free operation.

6 Future Work

Subsequently, we will look at how well this method works for more complex simulations with non-equidistant decision logic. We also want to extend our basic architecture to agent-based simulations that are not explicitly designed for use within an MDP. Another opportunity is to apply this non-parametric method with current policy gradient algorithms in the actor-critic design pattern. We expect it to be particularly well suited in the stochastic case.

References

- [1] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing Atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.
- [2] Sridhar Mahadevan and Georgios Theodorou. Optimizing production manufacturing using reinforcement learning. In *Proceedings of the Eleventh International Florida Artificial Intelligence Research Society Conference*, page 372–377. AAAI Press, 1998.
- [3] Bernd Waschneck, André Reichstaller, Lenz Belzner, Thomas Altenmüller, Thomas Bauernhansl, Alexander Knapp, and Andreas Kyek. Optimization of global production scheduling with deep reinforcement learning. *Procedia CIRP*, 72:1264–1269, 2018. 51st CIRP Conference on Manufacturing Systems.
- [4] Shuhui Qu, Jie Wang, and Govil Shivani. Learning adaptive dispatching rules for a manufacturing process system by using reinforcement learning approach. In *2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA)*, pages 1–8, Sep. 2016.
- [5] Stefanos Doltsinis, Pedro Ferreira, and Niels Lohse. An mdp model-based reinforcement learning approach for production station ramp-up optimization: Q-learning analysis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(9):1125–1138, Sep. 2014.
- [6] Laurent Capocchi and Jean-François Santucci. Discrete event modeling and simulation for reinforcement learning system design. *Information*, 13(3), 2022.
- [7] Sebastian Lang, Maximilian Kuetgens, Paul Reichardt, and Tobias Reggelin. Modeling production scheduling problems as reinforcement learning environments based on discrete-event simulation and openai gym. *IFAC-PapersOnLine*, 54(1):793–798, 2021. 17th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2021.
- [8] Andrew Greasley. Architectures for combining discrete-event simulation and machine learning. In *Proceedings of the 10th International Conference on Simulation and Modeling Methodologies, Technologies and Applications - SIMULTECH*, pages 47–58. INSTICC, SciTePress, 2020.
- [9] Benjamin Choo, Graham Crannel, Stephen Adams, Faraz Dadgostari, Peter A. Beling, Ann Bolcavage, and Roy McIntyre. Reinforcement learning from simulated environments: An encoder decoder framework. In *2020 Spring Simulation Conference (SpringSim)*, pages 1–12, May 2020.
- [10] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.
- [11] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: an Introduction*. The MIT Press, 2nd edition, 2018.

Simulation Technischer Systeme

Simulation of Technical Systems

Manöverbasierte Validierung einer Längsdynamiksimulation anhand generisch erzeugter Reallastverläufe (L)

S. Cortès, C. Dettmann, L. Heinzelmann, O. Haidar

Robustheitsanalyse einer Längsdynamiksimulation auf Basis verschiedener Qualitätskriterien (L)

S. Cortès, C. Dettmann, J. Heib, F. Millberg

Modellierung des Betriebsverhaltens eines Synchrongenerators als Stromerzeuger (L)

B. H. Herrmann, M. Ringel, D. Jörss, C. Fink

Numerische Simulation einer asymmetrischen Drallströmung - Vergleich der numerischen Turbulenzmodellierung mit experimentellen Messungen (K)

K. Zacharias, D. Welsch, W. Schlüter, A. Buchele

Systematischer modellbasierter Entwurf einer Reinforcement Learning-basierten neuronalen adaptiven Geschwindigkeitsregelung (L)

X. Liu-Henke, S. Jacobitz, Or Aviv Yarom, J. Fritz

Model-based Development of an Automated and Remotely-Controlled Demolition Excavator (L)

C. Richter, M. Shakoorianfard, V. Waurich, F. Will

Efficient state estimation for gas pipeline networks via low-rank approximations (L)

N. Stahl, N. Marheineke

Robotics simulation – A comparison of two state-of-the-art solutions (L)

M. Zwingel, C. May, M. Kalenberg, J. Franke

Parameter tables for PID controllers for time delayed systems optimized with a learning method (L)

R. Büchi

Interacting-Multiple-Model Partikelfilter zur Fehleridentifikation (L)

A. Lepp, D. Weidemann

Modelica simulation of pendulums and crane crab: PyMbs vs. OpenModelica (K)

M. Rößler, M. Ehrmann, F. Breitenecker

Die Langbeiträge (L) sind zu finden im **ARGESIM Report 20**,

Proceedings Langbeiträge ASIM SST 2022

ISBN ebook 978-3-901608-97-1, DOI: 10.11128/arep.20, ASIM Mitteilung AM 180

Numerische Simulation einer asymmetrischen Drallströmung - Vergleich der numerischen Turbulenzmodellierung mit experimentellen Messungen

Konstantin Zacharias^{1*}, Dennis Welsch¹, Wolfgang Schlüter¹, Alexander Buchele¹

¹Fakultät Technik, Hochschule Ansbach, Residenzstraße 8, 91522 Ansbach

*konstantin.zacharias@hs-ansbach.de

Abstract. Der Strömungszustand einer asymmetrischen Drallströmung tritt sehr häufig in Rohrleitungssystemen auf und stellt Durchflussmessgeräte vor die Herausforderung, die Messgröße trotz der gestörten Strömung korrekt zu bestimmen. Numerische Werkzeuge werden in der Durchflussmesstechnik immer häufiger verwendet und müssen in der Lage sein, diesen Strömungszustand exakt zu reproduzieren. In dieser Arbeit wird die asymmetrische Drallströmung mit drei unterschiedlichen numerischen Methoden der Turbulenzmodellierung simuliert und mit experimentellen Ergebnissen verglichen. Mit der skalen aufgelöste Delayed-Detached-Eddy-Simulation (DDES) kann die komplexe Strömungsstruktur mit einer hohen Genauigkeit und einem moderaten Rechenaufwand wiedergegeben werden.

Einleitung

In industriellen Anlagen werden Fluide über Rohrleitungen transportiert. Das Rohrsystem besteht in der Regel aus Rohrbögen, Ventilen, Pumpen und weiteren hydraulischen Komponenten. Alle diese Elemente stören den Strömungszustand und führen zu einem nicht voll ausgebildeten Strömungsprofil. Dies führt oft zu Schwierigkeiten bei der Durchflussmesstechnik, die ein ungestörtes Geschwindigkeitsprofil zur exakten Messung benötigt. Teststände für Durchflussmessgeräte müssen diese gestörten Strömungsbedingungen nachbilden können. Hierfür werden normierte Störkörper verwendet. Ein Rohrbogen verursacht einen Drall, der in einem Teststand über einen symmetrischen Drall-erzeuger nachgebildet werden kann. Dieser Störkörper besitzt jedoch eine schlechte Reproduzierbarkeit, da er sehr sensibel auf Anströmbedingungen reagiert [1]. Bereits kleine Störungen führen zu einem Bruch der Symmetrie. Díaz und Hinz [2] untersuchten die

Performance von Wirbelviskositätsmodellen bei Drallströmungen. Keines der untersuchten Turbulenzmodelle lieferte im Vergleich zum Experiment zufriedenstellende Ergebnisse. Ein Grund hierfür könnte die Sensibilität des Drallerzeugers auf geringfügige Störungen beim Einbau oder Fertigungstoleranzen sein. Tawackolian entwickelte einen neuen Störkörper, der einen asymmetrischen und drallbehafteten Strömungszustand erzeugt [3]. Dieser Strömungszustand wird im Rohrsystem durch einen Doppel-Rohrbogen verursacht. Der Störkörper, ein asymmetrischer Drallerzeuger, wurde von Turiso et al. [4] sowie Straka et al. [5] messtechnisch untersucht und als neuer Standardstörkörper zum Testen von Wasserzählern vorgeschlagen.

In dieser Arbeit wird die Nachlaufströmung eines asymmetrischen Drallerzeugers numerisch analysiert. Messtechnische Untersuchungen an drei Positionen [6] dienen als Validierung für die Strömungssimulationen. Ziel dieser Arbeit ist es drei unterschiedliche Methoden der Turbulenzmodellierung zu vergleichen und eine Auswahl hinsichtlich der Realitätsnähe zur Anwendung in der Industrie zu treffen.

1 Numerische Methoden der Turbulenzmodellierung

Für die numerischen Untersuchungen wird das kommerzielle CFD-Tool StarCCM+ von Siemens PLM Software verwendet. Nach dem Detaillierungsgrad der Turbulenz lassen sich die verwendeten numerischen Modelle in die Kategorien Wirbelviskositäts-, Reynoldsspannungsmodelle und der skalen aufgelösten-Modellierung unterteilen. Die Wirbelviskositätsmodelle unterliegen der Annahme der isotropen Turbulenz,

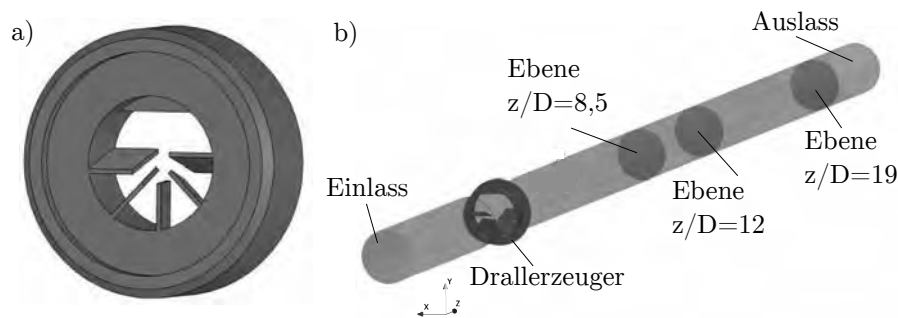


Abbildung 1: Asymmetrischer Drallerzeuger (a) und Simulationsdomäne mit Messebenen (b)

d. h. die Fluktuationen sind richtungsunabhängig. Als Wirbelviskositätsmodell wird das in der Industrie weit verbreitete $k-\omega$ -SST-Modell nach Menter verwendet. Es beinhaltet zwei zusätzliche Gleichungen für die kinetische Energie k und die spezifische Dissipationsrate ω zur Modellierung der Turbulenz. Im Reynoldsspannungsmodell (RSM) werden mit sechs zusätzlichen Gleichungen alle Komponenten des Reynoldsspannungstensors berechnet, weswegen das Modell insbesondere für Drallströmungen und anisotrope Turbulenzen geeignet ist. Das RSM hat einen höheren numerischen Aufwand und neigt zu Instabilitäten. In dieser Arbeit wird das überarbeitete Elliptic-Blending-RSM (EB-RSM) nach Lardeau und Menceau verwendet. Aufgrund der numerischen Stabilität wird die Lösung des $k-\omega$ -SST-Modell als Initiierung verwendet. Zur skalenaufgelösten-Modellierung zählt die Large-Eddy-Simulation (LES). Bei diesem numerischen Verfahren wird die Grobstruktur der Turbulenz direkt, d. h. ohne Modellierung berechnet und die statistisch einfach zu beschreibenden Feinstrukturen der Turbulenz modelliert. Dieses Verfahren benötigt ein sehr fein aufgelöstes Rechenggebiet und dementsprechend ist der numerische Aufwand sehr hoch. Eine Lösung bietet die Detached-Eddy-Simulation (DES). Sie ist ein hybrider Ansatz aus LES im relevanten Kernbereich und einem Wirbelviskositätsmodell in Wandnähe oder irrelevanten Strömungsgebiet. Die Rechenzeit ist für eine DES moderat und bietet somit eine Alternative für industrielle Anwendungen. Als skalenaufgelöste-Modellierung der Turbulenz wird die DDES-Methode gewählt, die als Wirbelviskositätsmodell das $k-\omega$ -SST-Modell verwendet. Die Tabelle 1 gibt eine Übersicht über die verwendeten numerischen Methoden der Turbulenzmodellierung und die Anzahl der Zellen. Für alle drei Simulationsmodelle wird eine Polyedervernetzung gewählt und

die Wandgrenzschicht auf einen dimensionslosen Wandabstand von $y^+ < 1$ aufgelöst. Abbildung 1 b) zeigt die komplette Simulationsdomäne mit der Einlaufstrecke, dem asymmetrischen Drallerzeuger, der Nachlaufstrecke und den drei Messebenen. Die Position des Drallerzeugers entspricht dem Experiment, mit der Blende parallel zur x-Achse ausgerichtet, die fünf Schaufeln im unteren und die Blende im oberen Bereich. Der Koordinatenursprung befindet sich direkt hinter der Blende im Rohr Mittelpunkt mit der z-Achse in Strömungsrichtung ausgerichtet.

	Rechenaufwand	Anzahl der Zellen
k-ω-SST	gering	3.5 Mio.
EB-RSM	gering	3.5 Mio.
DDES	moderat	12 Mio.

Tabelle 1: Übersicht der verwendeten numerischen Methoden der Turbulenzmodellierung

Für die Simulationen wird der Volumenstrom von $Q = 2,25 \text{ m}^3/\text{h}$ und eine Wassertemperatur von $T = 25^\circ\text{C}$, dem Experiment entsprechend, eingestellt. Die kinematische Viskosität beträgt $\nu = 0,893 \cdot 10^{-6} \text{ m}^2/\text{s}$ und die volumetrische Strömungsgeschwindigkeit $w_{vol} = 1,99 \text{ m/s}$. Dies entspricht einer Reynoldszahl von $Re = 4,5 \cdot 10^4$.

2 Ergebnisse

Der Vergleich der numerischen Turbulenzmodellierung mit den experimentellen Messungen findet an den drei Messebenen stromab des asymmetrischen Drallerzeugers statt. Abbildung 2 a) zeigt die normierten Konturplots der axialen Geschwindigkeitskomponente $\bar{w}/w_{vol}$

für die drei Messpositionen der LDA-Messung, des $k-\omega$ -SST-Modells, des EB-RSM und der DDES. Im Konturplot ist die Verlagerung des Strömungsschwerpunkts stromab des Drallerzeugers zu erkennen. Im Experiment (Abb. 2 a)) ist der Betrag der axialen Strömungsgeschwindigkeit stromab an der Position $z/D = 19$ am größten. Dies zeigt, dass die kinetische Energie der Sekundärströmung in die Hauptströmung transferiert wird. Die Strömungsstrukturen des $k-\omega$ -SST-Modells und des EB-RSM weisen in den Ebenen $z/D = 12$ und $z/D = 19$ große Ähnlichkeiten auf. Bei dem $k-\omega$ -SST-Modell ist an der Position $z/D = 8,5$ die Drallstruktur, die durch die Schaufeln verursacht wird, noch deutlich zu erkennen. Das EB-RSM besitzt an der Position $z/D = 8,5$ den höchsten Betrag der axialen Strömungsgeschwindigkeit und sinkt, anders als im Experiment, weiter stromab. Die Topologie des Dralls der Turbulenzmodelle ($k-\omega$ -SST-Modell & EB-RMS) unterscheidet sich im Vergleich zum Experiment. Jedoch ist die Position des Maximums der Strömungsgeschwindigkeit ansatzweise vergleichbar. Die DDES weist die größte Ähnlichkeit hinsichtlich des Betrags und der Topologie zur LDA-Messung auf. Der Betrag der axialen Strömungsgeschwindigkeit $\bar{w}$ ist bei der DDES geringfügig höher als beim Experiment, jedoch ist auch hier das Maximum in der Ebene $z/D = 19$ zwischen dem zweiten und dritten Quadranten erkennbar. Dies deutet darauf hin, dass der Drallwinkel der DDES mit dem Experiment übereinstimmt.

Abbildung 2 b) zeigt das normierte axiale Geschwindigkeitsprofil $\bar{w}/w_{vol}$ der Simulationen und der LDA-Messung, welches über die 10 Messpfade je Ebene gemittelt ist. Zum Vergleich ist das voll ausgebildete Geschwindigkeitsprofil nach Gersten und Herwig (G & H) abgebildet. Der Verlauf der Geschwindigkeitsprofile für das $k-\omega$ -SST-Modell und EB-RMS ist in den Ebenen $z/D = 8,5$ und $z/D = 19$ identisch. In der Ebene $z/D = 12$ zeigt das Profil der EB-RMS eine Überhöhung der Geschwindigkeit im Bereich $r/R > 0$ und eine kleinere Geschwindigkeit für den Bereich $-0.5 > r/R > 0$ im Vergleich zum $k-\omega$ -SST-Modell auf. Im Vergleich zum Experiment ähneln sich die Geschwindigkeitsverläufe der Turbulenzmodelle erst mit der Relaxation des Dralls ab der Ebene $z/D = 19$. Wie schon in den Konturplots zu erkennen, zeigt die DDES eine große Ähnlichkeit zum Experiment auf. Das gemittelte Geschwindigkeitsprofil der DDES und dem Experiment weist erst in der Ebene $z/D = 19$ im Bereich $r/R > 0$ eine leichte Divergenz

auf. In den Ebenen $z/D = 8,5$ und $z/D = 12$ stimmen die Profile aus Simulation und Experiment sehr gut überein. Der Betrag der Geschwindigkeit aus der DDES liegt systematisch über dem der Messung. Dies scheint ein Indiz dafür zu sein, dass der Massenstrom und somit die mittlere Strömungsgeschwindigkeit w_{vol} aus der Simulation und dem Experiment nicht exakt übereinstimmen.

3 Zusammenfassung und Ausblick

In dieser Arbeit wurden drei verschiedene Methoden der Turbulenzmodellierung auf ihr Potenzial zur Vorhersage einer asymmetrischen Drallströmung verglichen. Die Validierung erfolgte anhand experimenteller LDA-Messungen in drei Ebenen. Die Turbulenzmodelle ($k-\omega$ -SST-Modell & EB-RMS) zeigten vergleichbare Ergebnisse, obwohl das EB-RMS sechs zusätzliche Gleichungen für die Turbulenzmodellierung besitzt und für Drallströmungen geeigneter ist. Beide Modelle sind nicht in der Lage die Topologie des Dralls und den Geschwindigkeitsverlauf in den Ebenen exakt wiederzugeben. Die DDES hingegen simuliert die Struktur des Dralls sowie Geschwindigkeitsverlauf und -betrag mit hoher Genauigkeit. Aufgrund des moderaten Rechenaufwands und der hohen Genauigkeit eignet sich die DDES zur Berechnung komplexer Strömungsstrukturen in der Durchflussmesstechnik. In einer weiterführenden Untersuchung soll die Ursache für die Divergenz der Turbulenzmodelle genauer analysiert werden.

Literatur

- [1] Bück O. *Untersuchungen zur Darstellung und Weitergabe der Skala „Volumen von Wasser“ mithilfe laseroptischer und konventioneller Messverfahren* [Diss.]. Technische Universität Berlin; 2010.
- [2] Díaz O., Hinz D. Performance of eddy-viscosity turbulence models for predicting swirling pipe-flow: Simulations and laser-Doppler velocimetry. arXiv:1507.04648. 2015.
- [3] Tawackolian K. *Fluidodynamische Auswirkungen auf die Messabweichung von Ultraschall-Durchflussmessgeräten* [Diss.]. Technische Universität Berlin; 2013.
- [4] Turiso M., Straka M., Rose J., Bombis C., Hinz D. The asymmetric swirl disturbance generator: Towards a realistic and reproducible standard. *Flow Measurement*

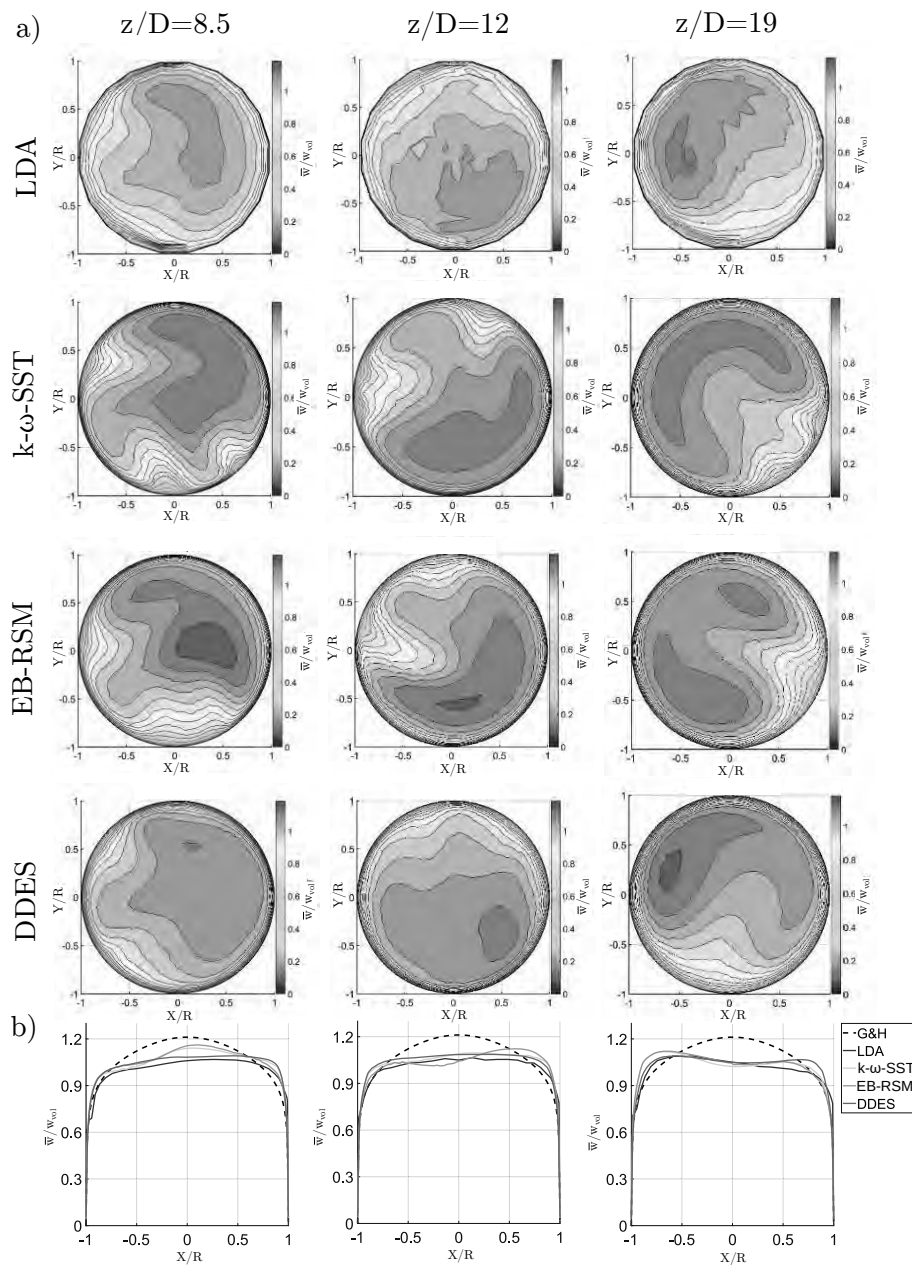


Abbildung 2: Konturplots (a) und gemittelt axiales Geschwindigkeitsprofil (b) an den drei Messpositionen der LDA-Messung und der verwendeten Methoden der Turbulenzmodellierung

and Instrumentation. 2018; 60:144-154. doi: <https://doi.org/10.1016/j.flowmeasinst.2018.02.021>.

- [5] Straka M., Eichler T., Koglin C., Rose J. Similarity of the asymmetric swirl generator and a double bend in the near-field range. *Flow Measurement and Instrumentation*. 2019; 70:101647. doi: <https://doi.org/10.1016/j.flowmeasinst.2019.101647>.

- [6] Welsch D., Zacharias K., Schlüter W. Experimental and numerical investigation of disturbed flow patterns by an asymmetric swirl generator. *3rd International Conference Business Meets Technology*; 2021 Sept; Valencia. doi: <https://doi.org/10.4995/BMT2021.2021.13595>.

Modelica Simulation of Pendulums and Crane Crab: PyMbs vs. OpenModelica

Martin Ehrmann¹, Matthias Rößler^{2*}, Felix Breitenecker¹

¹Institute for Analysis and Scientific Computing, TU Wien, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria

²dwh Simulation Services, Neustiftgasse 57-59, 1070 Vienna, Austria; *Matthias.Roessler@dwh.at

Abstract. Modelica has become an approved and comfortable tool for modeling. Modelica is a textual and graphical modelling environment with many application libraries. Modelica compilers are generating an implicit state space model (DAE model), which can be handled by simulation environments. This contribution presents a comparison between the two open software simulation environments PyMbs and OpenModelica for multibody systems using three case studies, a pendulum, a double pendulum, and a simplified version of a controlled crane crab. After description of the directly derived mathematical models and the setup of a model-specific error analysis for comparison purposes, implementations within PyMbs and OpenModelica are discussed. The final part consists of the numerical comparison of the results between the two simulation environments. Generally, not only deviations caused by the DAE algorithms can be observed, but also deviations caused by different DAE models generated by the Modelica compilers and caused by the stability of the systems.

Introduction

Classical multibody systems (MBS) and modern physical modelling systems as Modelica allow for comfortable modelling of multibody systems. Based on a high-level textual or/and graphical a-causal model description, these systems generate e.g. via Lagrange-mechanism the system governing ODE equations or DAE equations, resp. In order to compare results of Modelica-based simulations, it makes sense also to compare the results with results from the ‘basic’ ODE or DAE derived directly or ‘manually’.

The first case study is the *Classic Pendulum* (without friction). The law of conservation of energy results in the mathematical model equations:

$$\underbrace{I \frac{d^2\varphi}{dt^2}}_{= E_{kin}} = \underbrace{-\frac{1}{2} m g l \sin(\varphi)}_{= E_{pot}} \quad 0 = \ddot{\varphi} + \frac{3}{2} g \sin(\varphi)$$

The second case study is the *Double Pendulum*. It consists of two thin rods connected to each other with a revolute joint (frictionless) - Figure 1.

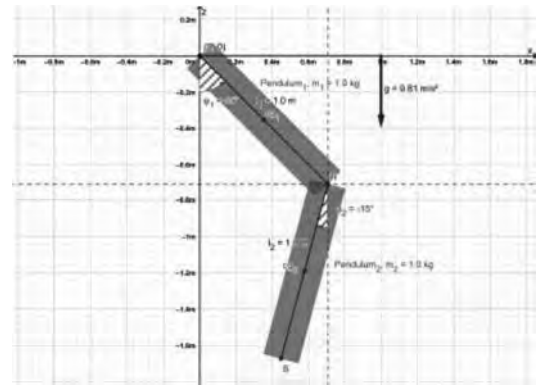


Figure 1: Schematic display of the *Double Pendulum* with parameter values.

Although a simple mechanical construct, the derivation of the system ODEs requires some formula mechanism. The Lagrangian for the specific parameters is

$$L = \frac{1}{6} (\dot{\varphi}_1^2 + 4\dot{\varphi}_1^2 + 3\dot{\varphi}_1\dot{\varphi}_2 \cos(\varphi_1 - \varphi_2)) + \frac{g}{2} (3\cos(\varphi_1) + \cos(\varphi_2))$$

Now derive the Lagrangian with respect to states and time properly, and the ODE equations of motion result:

$$\begin{aligned} 0 &= \frac{4}{3}\ddot{\varphi}_1 + \frac{1}{2} (\ddot{\varphi}_2 \cos(\varphi_1 - \varphi_2) + \dot{\varphi}_2^2 \sin(\varphi_1 - \varphi_2) + 3g \sin(\varphi_1)) \\ 0 &= \frac{1}{3}\ddot{\varphi}_2 + \frac{1}{2} (\ddot{\varphi}_1 \cos(\varphi_1 - \varphi_2) - \dot{\varphi}_1^2 \sin(\varphi_1 - \varphi_2) + g \sin(\varphi_2)) \end{aligned}$$

The third case study is the *Crane Crab* (or bridge crane). (Figure 2). Energy conservation f gives the Lagrangian

$$L = \frac{m_T + m_B + m_L}{2} \dot{x}^2 + \left(\frac{m_B}{6} + \frac{m_L}{2} \right) l^2 \dot{\varphi}^2 + \left(\frac{m_B}{2} + m_L \right) l \cos(\varphi) \dot{x} \dot{\varphi} + \left(\frac{m_B}{2} + m_L \right) l g \cos(\varphi)$$

Derivation of the Lagrangian with respect to states and time properly results in the ODE equations of motion:

$$\begin{aligned} (m_S + m_L) \ddot{x} + \mu \dot{x} + m_L l \ddot{\varphi} \cos(\varphi) - m_L l \dot{\varphi}^2 \sin(\varphi) - u &= 0 \\ l \ddot{\varphi} + g \sin(\varphi) + \ddot{x} \cos(\varphi) &= 0 \end{aligned}$$

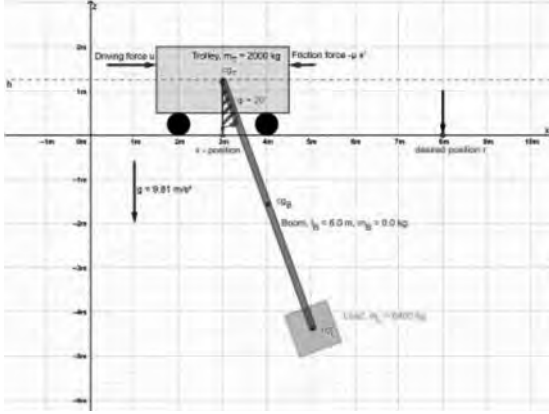


Figure 2: Schematic display of *Crane Crab* with parameter values.

The above equations make use of a simple velocity dependent friction model $\mu \cdot \dot{x}$ with a constant friction coefficient, used for the controller. For the nonlinear dynamics the model makes use of a piecewise continuous exponential friction model combined with a linear approximation for small velocities and adapted for independence of the direction of movement. This gives for higher velocities

$$F_C \cdot \text{sgn}(\dot{x}) + (F_{Brk} - F_C) \cdot e^{-(\dot{x}/c_v)^2} \cdot \text{sgn}(\dot{x}) + f \cdot \dot{x}$$

The results of the simulations are compared by means of the absolute differences of the results, in case of *Classic Pendulum* angle and angular velocity. The state vector consists in the above mathematical model of two angles and two angular velocities, in a coordinate system given in Figure 2. The Modelica environments under consideration make use of different states, so that for comparison here the coordinates of the tip must be used.

For case study *Crane Crab*, the differences of positions and of velocities can be used – and for simulation with controller, also the simulation time until target.

Furthermore, the time courses of the differences $\vec{w}(t_i)$ can by ‘summarized’ in norms, here in the Euclidean norm and in maximum norm:

$$\|w\|_2 = \left(\sum_{i=1}^{\text{length}(w)} w_i^2 \right)^{1/2} \quad \|w\|_\infty = \max_i |w_i|$$

1 PyMbs and OpenModelica Implementation

PyMbs [1] – developed at Dresden University of Technology published under the LGPLv3 licence – is a Python library with different options:

- (i) for generating standardized equations of motion files for multibody systems,
- (ii) to be simulated directly in the environment with graphical visualization (using NumPy),
- (iii) or to export the standardized model files for simulation in Python, MATLAB, Open Modelica, or C++.

The user makes use of a textual modelling environment combining masses with joints, etc. from object-oriented modules in libraries.

OpenModelica (OM; [2]) is a Modelica-based modelling and simulation environment first published in 2008 under an open source licence and is still under development. Open Modelica offers a textual and a graphical modelling environment, connecting blocks of libraries, here of the mechanics library, due to the Modelica standard.

OpenModelica comes along with a simulation environment, offering a wide variety of explicit and implicit ODE solvers, with implicit DASSL solver as default (BDF ODE solver).

The textual model implementations of the case study *Classic Pendulum* are similar in PyMbs and OM:

```
# create main object and initial reference frame
world = MbsSystem ([0, 0, -1])
# mass , length , inertia and body
m = world . addParam ('m', 1.0) .....
# add coordinate frames
pend . addFrame ('link', [0,0,-1/2.0])
pend . addFrame ('top', p=[0,0,0])
# add joint to constrain motion of the pendulum
revolute = world . addJoint (world , pend .top , 'Ry',
# generate equations of motion
world . genEquations . Explicit (), connecting
```

For case study *Double Pendulum*, the textual model implementation is similar to *Classic Pendulum*: two masses and two joints have to be connected, but the amount of parameter definitions is growing. And the choice of the coordinate system is not unique, Easier is to use is OM’s graphical modelling environment, see Figure 3.



Figure 3: Double Pendulum - OpenModelica (OM) block representation.

The textual model implementation of case study *Crane Crab* is straightforward, only the implementation of the nonlinear friction and of the state controller requires additional programming, below a code snippet:

```
# add controller
G=world.addController('G',controlForceFrc,shape=(3,))
world.addLoad.CmpForce(G,trolley,world,
name='DrivingForce')
.....
k=[176146.79,-263282.49,204004.59,-1065027.52]
F=-np.dot(y,k)+v*r-friction(y[2])
```

OM's graphical modelling interface allows for easier modelling of the mechanical parts – only two masses, and a revolution joint and a sliding joint, but the implementation of the controller requires elaborate signal-oriented graphical modelling (Figure 4).

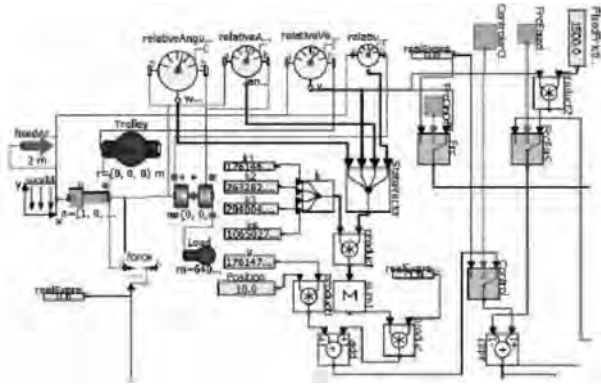


Figure 4: *Crane Crab* - part of the OpenModelica block representation with control.

2 PyMbs and OpenModelica Simulation Results

The case study *Classic Pendulum* compares four approaches:

- modelling in PyMbs and simulation in PyMbs via NumPy
- modelling in PyMbs, model export to OM, simulation in OM –
- modelling in OM and simulation in OM
- use of mathematical equations and simulation with standard ODE solver (MEq)

For appropriate comparisons of numerical results, similar (implicit) ODE solvers with similar accuracy tuning are used. The deviations for *Classic Pendulum* increase with bigger initial values – see Figure 5 and Table 1, for analysing PyMbs vs. OM.

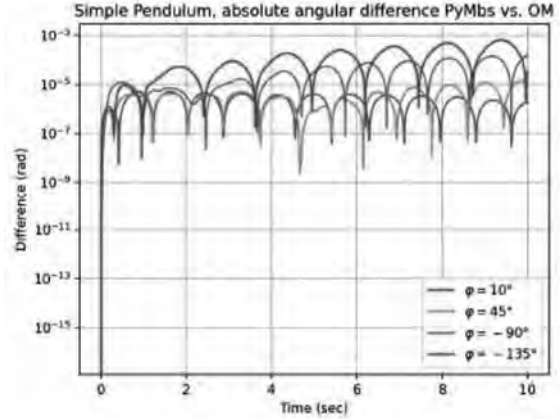


Figure 5: *Classic Pendulum* - Absolute difference between the angles, PyMbs vs. OM.

initial value	$\ \cdot\ _2$ norm	$\ \cdot\ _\infty$ norm
$\varphi = 10^\circ$	$8.39097 \cdot 10^{-5}$	$5.19584 \cdot 10^{-6}$
$\varphi = 45^\circ$	$1.46363 \cdot 10^{-4}$	$1.34116 \cdot 10^{-5}$
$\varphi = -90^\circ$	$1.62884 \cdot 10^{-3}$	$1.51452 \cdot 10^{-4}$
$\varphi = -135^\circ$	$6.60355 \cdot 10^{-3}$	$6.46902 \cdot 10^{-4}$

Table 1: *Classic Pendulum* – norm deviation for PyMbs vs. OM.

The comparison of OM vs. MEq indicates a high accuracy of OM solution, Figure 6 and Table 2.

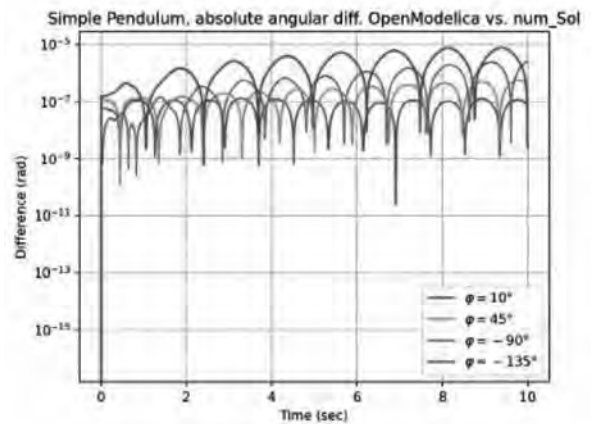


Figure 6: *Classic Pendulum* - absolute angle difference OM vs. MEq.

initial value	$\ \cdot\ _2$ norm	$\ \cdot\ _\infty$ norm
$\varphi = 10^\circ$	$2.58719 \cdot 10^{-6}$	$1.32432 \cdot 10^{-7}$
$\varphi = 45^\circ$	$7.56173 \cdot 10^{-6}$	$5.61011 \cdot 10^{-7}$
$\varphi = -90^\circ$	$2.73603 \cdot 10^{-5}$	$2.59398 \cdot 10^{-6}$
$\varphi = -135^\circ$	$1.00597 \cdot 10^{-4}$	$7.76396 \cdot 10^{-6}$

Table 2: *Classic Pendulum* – norm deviation for OM vs. MEq.

The smallest deviations occur comparing OM vs. PyMbs /OM – same numerics is used.

The case study *Double Pendulum* results in deviations reaching 10^{-1} – here shown in Figure 7 and Table 3 for case PyMbs vs. OM. (using tip position as variable).

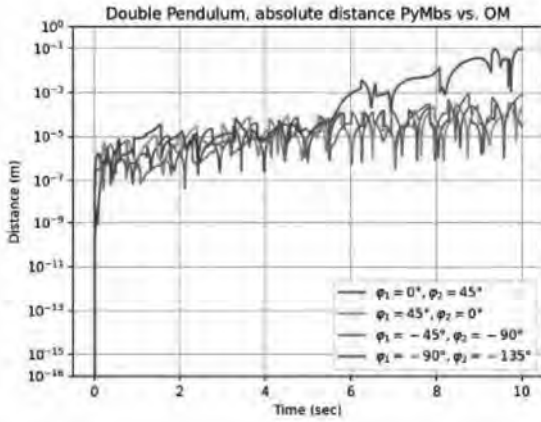


Figure 7: *Double Pendulum* - absolute difference between tip positions, PyMbs vs. OM.

initial value φ_1	initial value φ_2	$\ \cdot \ _2$ norm	$\ \cdot \ _\infty$ norm
$\varphi_1 = 0^\circ$	$\varphi_2 = 45^\circ$	$2.14831 \cdot 10^{-3}$	$3.07657 \cdot 10^{-4}$
$\varphi_1 = 45^\circ$	$\varphi_2 = 0^\circ$	$2.08194 \cdot 10^{-3}$	$3.55829 \cdot 10^{-4}$
$\varphi_1 = -45^\circ$	$\varphi_2 = -90^\circ$	$4.87876 \cdot 10^{-3}$	$8.95349 \cdot 10^{-4}$
$\varphi_1 = -90^\circ$	$\varphi_2 = -135^\circ$	$5.84929 \cdot 10^{-3}$	$9.79425 \cdot 10^{-4}$

Table 3: *Double Pendulum* – norm deviation for PyMbs vs. OM.

In the case study *Crane Crab*, the controller results in a better numerical stability, so that deviations are relatively small and compensated, here again case PyMbs vs. OM in Figure 8 and Table 4.

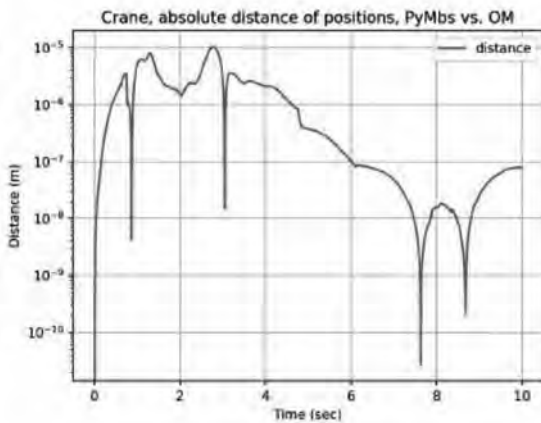


Figure 8: *Crane Crab* - absolute difference between positions, PyMbs vs. OM.

environment	position $x(t)$ at time step t	
	$x(10.0)$	$ 10.0 - x(10.0) $
PyMbs	10.00349 m	$3.48552 \cdot 10^{-3}$ m
OpenModelica	10.00349 m	$3.48544 \cdot 10^{-3}$ m

Table 4: *Crane Crab* - deviation of target position, PyMbs vs. OM.

3 PyMbs vs OpenModelica - Summary

Setting up the PyMbs library is rather complex because it is dependent on very specific. Once the library is set up, the actual modelling of the simulations as well as the process of changing simulation parameters is convenient - everything has to be written in Python and can be changed by simple flag-variable and/or the usage of comments. The portability of the PyMbs model can be considered very high, as long as no external functions for the controller and/or friction in case study *Crane Crab* are used).

OpenModelica is easier to set up and offers the graphical modelling environment. On the other hand, use of blocks for control is cumbersome. In general, OM shows advantages because of a long development time.

The comparisons of accuracies are dangerous, because they depend on numerical tuning parameters. But case studies *Classic Pendulum* and *Double Pendulum* show a tendency for less accuracy of PyMbs (vs. OM and MEq); and interestingly, accuracy of OM and PyMbs/OM are almost identical (here indeed the same ODE solvers are used, and obviously similar model descriptions). Case study *Crane Crab* also shows this tendency, but the control compensates the differences in accuracy.

It must be noted, that the conclusions hold for the case studies, and are not general ones – but they show a tendency. Further comparisons of these case studies are on work, especially with MEq results for case studies *Double Pendulum* and *Crane Crab*.

References

- [1] PyMbs Team. PyMbs Download. Accessed: 2022-02-17. url: <https://bitbucket.org/pymbs/pymbs/src/master/>.
- [2] OpenModelica Documentation. OpenModelica Users Guide Solving Methods. Accessed: 2022-02-17. url: <https://www.openmodelica.org/doc/OpenModelicaUsersGuide/latest/solving.html>

Simulation Biologie, Medizin und Umwelt

Simulation in Biology, Medicine, and Environment

Ein Fitnesscoach für Menschen mit Querschnittslähmung (L)

K. Nickel, J. Bolz

Wirkstoffidentifikation anhand mikrobieller Proteinmuster (K)

O. Al-Latif, H. Hillenhagen, J. Bernhardt, A. Grüning, C. Wahmkow

Die Flugrouten der Graukraniche – eine GIS-basierte Analyse zum Zusammenhang zwischen Routenwahl und Wetterbedingungen (K)

S. Schreiber, T. Christiansen, J. Wittmann

Modelling Blood Volume Oscillations in the Finger (K)

H. Kastinger, M. Bachler

SIR-type Epidemic Models for Education at Web (K)

A. Freißlinger, A. Körner, F. Breiteneker

Die Langbeiträge (L) sind zu finden im **ARGESIM Report 20,**
Proceedings Langbeiträge ASIM SST 2022

ISBN ebook 978-3-901608-97-1, DOI: 10.11128/arep.20, ASIM Mitteilung AM 180

Wirkstoffidentifikation anhand mikrobieller Proteinmuster

Omar Al-Latif¹, Heiko Hillenhagen¹, Jörg Bernhardt², André Grüning³,
Christine Wahmkow^{1*}

¹Fakultät für Maschinenbau, Hochschule Stralsund, Zur Schwedenschanze 15
18435 Stralsund, Germany; **Christine.Wahmkow@hochschule-stralsund.de*

²Institut für Mikrobiologie, Universität Greifswald

³Institute for Applied Computer Science (IACS), Hochschule Stralsund, Zur Schwedenschanze 15

Abstract. Dargestellt wird ein Gemeinschaftsprojekt der Hochschule Stralsund mit der Universität Greifswald. Am dortigen Institut für Mikrobiologie werden durch Einsatz der Protein-Massenspektroskopie Reaktionen von Mikroorganismen während diverser Wirkstoffbehandlungen auf der Ebene zellulärer Proteinmuster untersucht. Es entstehen ausreichend große Datensätze, die für die Untersuchungen mit Methoden der Künstlichen Intelligenz geeignet sein sollten. Ziel der vorliegenden Studie war, das Proteinmuster eines unbekannten Wirkstoffes jenen bekannter Wirkstoffe zuzuordnen und so eine Klassifizierung zu ermöglichen. Die Ergebnisse sollen verglichen werden mit den auf statistischen Methoden basierenden Ergebnissen des Institutes.

Einleitung

Medikamente- insbesondere Antibiotika sollen krankheitsauslösende Bakterien bekämpfen. Um die Wirkmechanismen innerhalb der Bakterien zu verstehen, erfassen Mikrobiologen die Änderungen zellulärer Proteinmuster in Bakterienzellen in Reaktion auf die Wirkstoffzugabe. Die ermittelten auf massenspektrometrischen Proteindaten basierende Datensätze sind zwar groß, jedoch nicht homogen. Messfehler, technische Grenzen der Massenspektroskopie (MS) und spontane Fluktuationen in biologischen Systemen manifestieren sich in den Datensätzen. Dies steht einer direkten Verwendung für KI-Modelle im Wege. Eine aufwändige Datenvorverarbeitung bewegt sich jedoch auf einem schmalen Grat. Bei zu starker Bereinigung verbleiben zu wenig Daten für die Ableitung signifikanter Modelle, zu viele besonders schlecht reproduzierbare Daten verfälschen ebenfalls das Ergebnis.

1 Datenerhebung

Wird ein Organismus einem Wirkstoff ausgesetzt, so reagiert dieser mit der An- und Abschaltung von Genen für betroffene zelluläre Strukturen und Funktionen. Geschädigte Proteine müssen ersetzt werden, es kommen "frische" Proteine hinzu, unnütze Proteine werden abgebaut und verschwinden. Wirkstoffspezifisch werden auch völlig neue Proteine gebildet. Kurz: Die Gesamtheit der Proteine verändert sich in den relativen Proteinmengen im zeitlichen Verlauf der Wirkstoffapplikation.

Für die mikrobiologische Untersuchung wird eine Bakterienkultur in einem Nährmedium herangezogen und einem Wirkstoff ausgesetzt. Um die Änderungen der Proteinmengen zu erfassen, wird zu verschiedenen Zeiten eine Probe entnommen. Per MS werden die Massen und die Mengen der Proteine aufgezeichnet. Um biologische und technische Schwankungen auszugleichen, werden für jede Wirkstoffbehandlung mehrere biologische Replikate erstellt, aus denen zu jedem Messzeitpunkt jeweils drei Proben (technische Replikate) entnommen werden. Als Kontrollexperiment fungieren unbehandelte Proben. [1]

Die Abbildung 1 verdeutlicht exemplarisch die Probenahme: Aus unabhängigen Vorkulturen werden drei Bioreplikate angezogen, welche nicht behandelt (Kontrolle) oder mit verschiedenen Wirkstoffen mit bekannten als auch unbekannten Wirkmechanismus behandelt werden. Zu verschiedenen Zeitpunkten (30, 60, 90 und 120 Minuten) werden jeweils 3 technische Replikate vermessen. Das ergibt 9 Proben pro Zeitpunkt und (Nicht)Behandlung. Nach dem beschriebenen Messvorgang sind für 9 Wirkstoffe je 3451 Proteine vermessen worden. Jedes Protein besitzt in einer idealtypischen

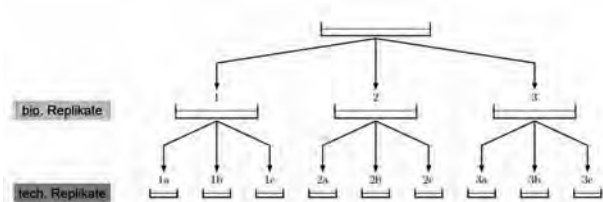


Abbildung 1: Beispielhafte Darstellung der Datenerhebung und Datenbeschriftung für einen Wirkstoff zu einem Zeitpunkt. Es ergeben sich für jeden Wirkstoff 9 Proben pro Zeiteinheit.

Matrix 450 Datenpunkte. Diese Zahl ist die Summe aller Messpunkte aller Wirkstoffe. Aus technischen Gründen sind jedoch nicht alle Messreihen vollständig.

2 Das Problem fehlender Daten

Von den insgesamt 3451 verschiedenen Proteinen konnten nur für 298 die kompletten Daten über alle Zeitpunkte und Wirkstoffe gemessen werden. Es gibt 2 Ursachen für die Datenlücken. 1. Ist die Menge bestimmter Proteine zu klein, wird die Messschwelle nicht erreicht - es wird kein Messwert ermittelt. 2. Fluktuiert die Proteinmenge um diese Messschwelle, dann werden nur oberhalb der Messschwelle Daten ermittelt. Die Zeile in der Datenmatrix wird lückenhaft. Es stellt sich also die Frage, wie in diese idealtypische Matrix mit einer Dimension von 3451×450 Datenpunkten sinnvoll fehlende Messwerte imputiert werden können. Die Art der fehlenden Daten beschreibt deren Unvollständigkeit in dem Sinne, dass deren Fehlen sowohl von ihrer Ausprägung selbst als auch von anderen Faktoren abhängig ist. Diese Kategorie wird *Missing Not At Random* (MNAR) bezeichnet.[2]

Zur Vervollständigung der Daten wird auf mathematische Verfahren der Statistik zurückgegriffen. Die Software R bietet dazu ausreichend Möglichkeiten. Bevor in R die Imputation erfolgen kann, ist es wichtig, den gesamten Datensatz so aufzuteilen, dass der Zahlenbereich, aus dem die Werte ermittelt werden, auch plausibel ist. Eine Aufteilung der Gesamtmatrix nach den einzelnen Wirkstoffen führt zu 9 Matrizen der Größe 3451×54 . Hier wird der Zusammenhang der einzelnen Datenpunkte innerhalb der Wirkstoffe in Form der zeitlichen Verläufe der Proteinmassen sichtbar.

Die Aufteilung der Gesamtmatrix in die einzelnen Wirkstoffe verkürzt die Anzahl der Eingangsdaten er-

heblich. Sie ist jedoch noch immer für einen effizienten Imputationsdurchlauf zu hoch. Mit steigender Anzahl fehlender Werte wird die Qualität der Imputationsergebnisse stark nachlassen. Deshalb ist der Versuch, fehlende Werte einer Zeile zu ermitteln, die nur einen einzigen Wert enthält, nicht zielführend. Selbst wenn eine Zeile zur Hälfte mit Werten gefüllt ist, ist die Chance für unrealistische Werte und Zufallszahlen bei einer Imputation zu hoch. Eine realistische Schranke für die Menge fehlender Werte, bei der die Imputation noch gute Ergebnisse erzielen kann, liegt bei etwa 25 Prozent fehlender Datenpunkte. Aus diesem Grund wird für jeden Wirkstoff ein neuer Datensatz erstellt, in den alle Proteinzeilen exportiert werden, die oberhalb dieser Schranke liegen, folglich über 75 Prozent vorhandene Werte besitzen. Zusätzlich werden in einem weiteren Datensatz aus dem Original alle Proteinzeilen exportiert, die über 80 Prozent vorhandene Werte haben. Auf diese Weise entstehen zwei Datensätze für jeden Wirkstoff. Da diese Datensätze auch 100 Prozent Messwerte enthalten können und dafür eine Imputation unnötig ist, werden diese Datensätze ebenfalls ausgesondert. Abbildung 2 zeigt das Vorgehen für alle Wirkstoffe. Beispielhaft ist Wirkstoff 5 fett hervorgehoben. Es

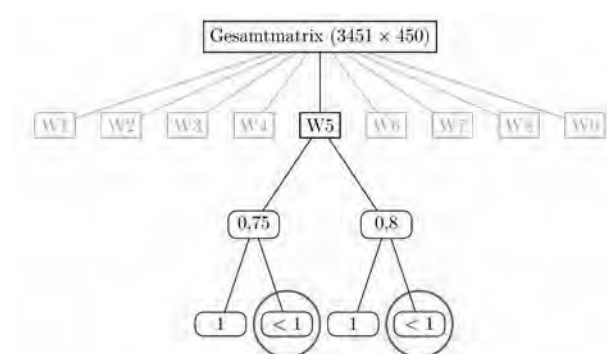


Abbildung 2: Darstellung der Datenaufbereitung für die Imputation

wird das Verfahren der *Multiplen Imputation* angewendet, bei der iterativ die passenden Werte ermittelt werden, die Vorgehensweise wird *Predictive Mean Matching* (kurz: PMM) genannt und ist in [3] beschrieben. Nach der Validierung der imputierten Werte kann von einer guten Qualität ausgegangen werden, ohne sicher zu sein, dass es die biologisch richtigen Daten sind. Durch die Imputation entsteht im Durchschnitt eine 4-fache Datenmenge der lückenhaften Ausgangsmenge aus vollständigen Messungen. Die Abbildung 3 zeigt

beispielhaft den zeitlichen Proteinmassenverlauf eines Proteins. Dabei ist nicht nur ersichtlich, dass die Entwicklung über die verschiedenen Messungen entscheidend ist. Es zeigt auch, dass die unterschiedlichen biologischen Replikate, die hier in unterschiedlichen Farben dargestellt sind, auch voneinander abweichen.

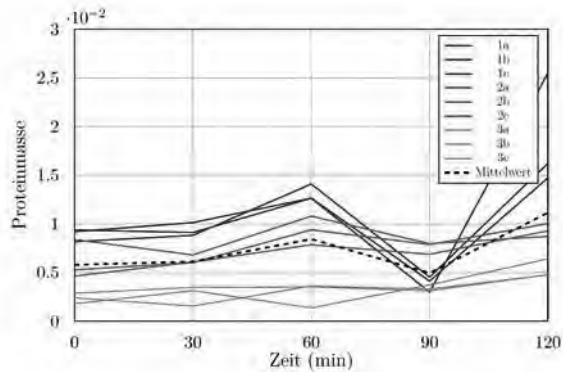


Abbildung 3: Beispielhafter Verlauf des Proteins dnaA für den Wirkstoff Tetracyclin (Tet).

3 Modellbildung

Für die Modellbildung stehen verschiedene Tools zur Verfügung. Hier im Projekt wurden die MatLab-Tools wie auch das auf Python basierende Scikit-Learn Kit benutzt. Die vorverarbeiteten Daten mit Imputationen lieferten die Grundlagen zur Nutzung verschiedener Modelle.

Ergebnisse mit Matlab-Tools. Von den 4 möglichen Applikationen der Matlab Deep Learning Toolbox wurden die Apps *Neural Net Clustering* sowie *Neural Network Pattern Recognition* genutzt.

Die Clustering App durchsucht die Eingangsdaten nach Mustern und versucht, die Daten in Klassen einzuordnen. Sie kann zwar nicht zielgenau einen Wirkstoff wiedererkennen, liefert jedoch zumindest Datengruppen, in die die Wirkstoffe eingeordnet werden können. Dieses unüberwachte Lernverfahren kann zur Erstellung von Topografien herangezogen werden, um einen Überblick über eine Datenstruktur zu erhalten. Der Algorithmus der Self-Organizing-Map (SOM) verändert die Gewichtsvektoren der Neuronen entsprechend der Hebb'schen Lernregel, sodass sich im Laufe ausreichender Iterationen die Netzstruktur als eine topografische Abbildung beschreiben lässt. Es ist so gesehen ein Werk-

zeug zur Visualisierung von hochdimensionalen Daten.[4]

Der Vorteil der App *Neural Network Pattern Recognition* liegt in der Vordefinition von Klassen und ist damit ein Verfahren des überwachten Lernens. Bekannte Wirkstoffe können dem Netz direkt zugänglich gemacht werden, indem man diese per Definition einpflegt. Neu gefundene Klassen könnten nach jedem Vorgang zur initialen Definition hinzugefügt werden. Durch das Hinzufügen von immer mehr Zielklassen könnte jedoch die Genauigkeit des Netzes verringert werden, da mit steigender Anzahl an Datenpunkten die Suche nach Mustern erschwert wird.

Die Eingangsmatrix ist dieselbe wie bei der Anwendung der Clustering App. Nur muss neben ihr für eine erfolgreiche Anwendung der Pattern Recognition App eine Zielmatrix mit angegeben werden. Das ist die Angabe des gegebenen Wirkstoffes auf eine Substanz.

Die Confusion-Matrix als eine grafische Aufbereitungsmethode dieser App zeigt die erreichte Genauigkeit der jeweiligen Netzdurchläufe. Beispielhaft ist in Abbildung 4 eine solche Matrix für den Datensatz mit 75-% vorhandenem Datenvolumen gezeigt. Auffallend ist,

Confusion Matrix D _{0.75}									
Output Class	1	2	3	4	5	6	7	8	9
1	36 8.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%
2	0 0.0%	36 8.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%
3	0 0.0%	0 0.0%	54 12.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%
4	10 2.2%	0 0.0%	0 0.0%	44 9.8%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%
5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	54 12.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%
6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	54 12.0%	0 0.0%	0 0.0%	0 0.0%
7	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	54 12.0%	0 0.0%	0 0.0%
8	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	54 12.0%	0 0.0%
9	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	54 12.0%
Target Class	78.3%	100%	100%	100%	100%	100%	100%	100%	97.8%
	21.7%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	0.0%	2.2%

Abbildung 4: Darstellung einer Confusion-Matrixes für den Datensatz D0.75 mit 75%-igem Datenvolumen.

dass der Netzfehler der Datensätze bei Anwendung anderer Streuungen an den gleichen Stellen, nämlich bei der 1. und 4. Klasse, d.h. beim 1. und 4. Wirkstoff auf-

tritt. Eine weitere Analyse mit mehr Daten könnte das Problem aufklären.

Ergebnisse mit dem Python-Tool Scikit-Learn.

Für die Tests wurde in [5] der *MLPClassifier* basierend auf Neuronalen Netzen gewählt. Als Lernalgorithmus liegt Backpropagation zugrunde. Netztopologien mit einigen wenigen bis zu mehreren tausend Neuronen in einer von bis zu 3 verdeckten Schichten gingen in die Untersuchung ein. Die Netzarchitektur bestehend aus zwei oder drei verdeckten Schichten mit jeweils 400-600 Neuronen ist über alle Versuche hinweg die Gruppe an Netzen, die die besten Ergebnisse brachte. Die "großen" Tausender-Netze konnten nur selten bessere Ergebnisse bringen. Da die Klassifizierungsmethoden insgesamt letztendlich nicht die erhofften nachvollziehbaren Ergebnisse gebracht haben, wurde in einem nächsten Schritt der Algorithmus der *Principal Component Analysis* zur Dimensionsreduzierung implementiert. Durch die Projektion von Merkmalsverteilungen auf Ebenen wird die Anzahl der Merkmale zwar reduziert, aber die Genauigkeit des Modells kann bei richtiger Auswahl der zu projektierenden Merkmale erhöht werden. Grafisch dargestellt werden können die Merkmalsverteilungen in Form von Dendrogrammen oder auch in 3-dimensionalen Strukturen zur visuellen Veranschaulichung. Abbildung 5 zeigt einen solchen Würfel, in dem die Klassen gut erkennbar sind. Es wurde von einem ca. 3000-dimensionalen Raum auf 3D reduziert. Vergleichbar zu den Ergebnissen, die mittels Statistiksoftware R in Greifswald erzielt wurden, ist ein Dendrogramm. Es stellt ein Baumdiagramm dar und dient der Visualisierung einer Clusteranalyse. Objekte werden nach Ähnlichkeit der Merkmalsausprägung zu Clustern gruppiert. Das beste Ergebnis ist dem Dendrogramm der Greifswalder sehr ähnlich. Beide Dendrogramme werden auf der Tagung im Vortrag gezeigt.

Zusammenfassung

Theoretisch lag mit ca. 750 000 Messwerten eine ausreichend große Datenmenge zur Auswertung vor. Es kommt dabei auf die Form an, in der die Daten existieren. Wichtig ist, dass es eine ausreichende Anzahl an tatsächlichen Messwerten gibt. Auf diese Weise erhält ein Modell des Maschinellen Lernens viele Beispiele, anhand derer es Muster ermitteln kann. Die vorhandene Datenbasis enthielt nur ca. 50% tatsächliche Messwerte. Demzufolge war die Qualität und Quantität

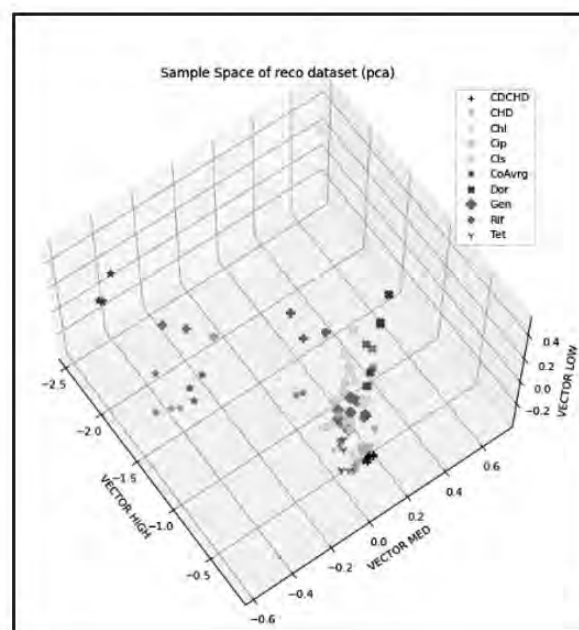


Abbildung 5: Darstellung des 3D-Raumes zur Dimensionsreduzierung von ca. 3000D.

der Daten für sichere Aussagen unzureichend. Nach einer aufwändigen Datenimputation konnten Merkmalsausprägungen bei Proteinveränderungen für die unterschiedlich eingesetzten Wirkstoffe erkannt und dargestellt werden.

Literatur

- [1] Hillenhausen H. *Entwicklung Neuronaler Netze zur Proteinidentifikation* [Bachelorarbeit]. [Fakultät für Maschinenbau]. Hochschule Stralsund; 2020
- [2] Oliver Lüdtker u. a. „Umgang mit fehlenden Werten in der psychologischen Forschung“. In: Psychologische Rundschau 58.2 (2007), S. 103–117.
- [3] Stef Van Buuren. 3.4 Predictive mean matching. url: <https://stefvanbuuren.name/fimd/sec-pmm.html> (besucht am 06. 04. 2020).
- [4] Klammer, R. *Alternative Analysemöglichkeiten geographischer Daten in der Kartographie mittels Self-Organizing Maps* [wissenschaftlicher Beitrag]. [Institut für Kartographie]. Technische Universität Dresden ; 2010
- [5] Vajen, R. *Entwicklung Neuronaler Netze zur Wirkstoffidentifikation in der Mikrobiologie* [Masterthesis]. [Fakultät für Maschinenbau]. Hochschule Stralsund; 2020

Daher ist im dicht besiedelten und genutzten Raum ein nachhaltiges Management notwendig, das die Interessen sämtlicher Shareholder, also der ziehenden Kraniche wie der betroffenen Landnutzung berücksichtigt.

Als Grundlage eines solchen Managements ist das Verständnis und im zweiten Schritt vielleicht sogar eine Prognose des kurz- und mittelfristigen Zugverhaltens der Tiere von entscheidender Bedeutung. Das Zugverhalten wiederum wird maßgeblich durch die aktuellen Witterungsbedingungen beeinflusst.

Daher wurden in diesem Projekt für den Raum der BRD, durch den Teile der westeuropäischen Zugroute hindurchführen, Untersuchungen bezüglich des Einflusses des Wetters auf das Flugverhalten angestellt. Dabei kann auf detaillierte, öffentlich zugängliche Wetterdaten des Deutschen Wetterdienstes sowie auf GPS-Daten von getrackten Kranichen des Kranichschutz Deutschland zugegriffen werden, die im folgenden Abschnitt näher spezifiziert werden. Ziel der Arbeit soll es sein, die Bewegung von Kranichen und in Abhängigkeit von den Veränderungen der Wetterbedingungen zu visualisieren, um festzustellen, in welcher Form die Tiere ihre Flugroute anpassen und ob eine solche Flugroutenanpassung für den Zug in den Folgejahren die ursprüngliche Flugroute ersetzt.

2 Verwendetes Datenmaterial

Grundlage aller Visualisierungen und Auswertungen sind die GPS-Daten von getrackten Tieren. Die Kraniche werden vom Kranichschutz Deutschland mit GSM-GPS-Sendern der Firma e-obs GmbH ausgestattet, die die aktuelle Position via SMS täglich an das NABU-Kranichzentrum in Groß Mohrdorf schickt. Enthalten sind jeweils fünf GPS-Positionen sowie die Neigung von drei Achsen die weiteren Aufschluss über das Verhalten der Tiere geben können (vgl. [1], Besenderung). Die Ausstattung der Sensoren wird durch Patenschaften finanziert. Für die Weiterverarbeitung im GIS dient eine Excel-Datei, die im Wesentlichen die aufgezeichneten Punkte des Tracks in Form von Breitengrad, Längengrad und Zeitstempel enthält.

timestamp	Location-long	Location-lat	Species-type	Tag	local-identifier	individual-identifier	local-identifier	Study-name
2013-05-19 08:00:07.000	13.116778746714795	53.56733345208416	gms		3123	3123	Kranichname	Gms gms project 2013
2013-05-19 11:00:07.000	12.6480888421058	53.4452997629473	gms		3123	3124	Kranichname	Gms gms project 2013
2013-05-19 11:00:34.000	12.18879894479522	53.12547086254602	gms		3123	3125	Kranichname	Gms gms project 2013
2013-05-19 09:23.000	12.25492099465278	53.2546627188279	gms		3123	3126	Kranichname	Gms gms project 2013
2013-05-20 05:00:07.000	12.33495934543297	53.37836791287912	gms		3123	3127	Kranichname	Gms gms project 2013
2013-05-20 06:00:06.000	12.434345676818873	53.48823492586172	gms		3123	3128	Kranichname	Gms gms project 2013
2013-05-20 11:00:12.000	12.531123212317827	53.58673446267475	gms		3123	3129	Kranichname	Gms gms project 2013
2013-05-20 13:00:18.000	12.632215878787878	53.632215878787878	gms		3123	3130	Kranichname	Gms gms project 2013
2013-05-21 04:00:23.000	12.78234323073881	53.75286752873246	gms		3123	3131	Kranichname	Gms gms project 2013
2013-05-21 06:00:07.000	12.88732488232817	53.82347627364354	gms		3123	3132	Kranichname	Gms gms project 2013
2013-05-21 08:00:07.000	12.940973394672323	53.81287237848712	gms		3123	3133	Kranichname	Gms gms project 2013
2021-05-21 11:00:28.000	12.15289862478623	53.1051281672537	gms		3123	3134	Kranichname	Gms gms project 2021

Abbildung 2: Ausgangsdaten Kranich-Tracking

Die Abbildung 2 zeigt das Format der Kranichdaten. Die Werte wurden jedoch verfälscht, da eine explizite Veröffentlichung der exakten GPS- und Zeitdaten der Kraniche aus Schutzgründen nicht erlaubt ist. Tracking-Daten waren verfügbar im Zeitraum Sommer2013 bis zum Frühjahr 2021. Die Daten konnten ohne weitere Umformatierung und Vorverarbeitungsschritte direkt in ArcGIS importiert werden und können dort sowohl räumlich als auch im zeitlichen Verlauf dargestellt werden. Für jedes Jahr existiert eine eigene Datei, die auch so in das ArcGIS Pro eingepflegt wurde, um die Möglichkeit zu haben, sich die einzelnen Jahre separat anzusehen und zu analysieren. Es gibt täglich 5 GPS-Daten, die zwischen 3 und 13 Uhr aufgezeichnet sind. Es gibt keine wesentlichen zeitlichen Lücken, lediglich vereinzelt fehlen Tage.

Die entsprechenden Wetterdaten konnten aus dem Climate-Data-Center (CDC) [2] des Deutschen Wetterdienstes (DWD) entnommen werden. Dort werden sämtliche Parameter zur Verfügung gestellt, die einen Einfluss auf das Zugverhalten erwarten lassen: Bodentemperatur, Niederschlag, Windrichtung und Windgeschwindigkeit. Für diese Parameter liegen Messwerte für alle des Deutschen Wetterdienstes veröffentlichten Messpunkte in Deutschland vor. Die Messwerte liegen zeitlich mindestens täglich vor. Für die Bodentemperatur und die Windrichtung sogar stündlich. Es wurden für die Untersuchung die genaueste zeitliche Auflösung gewählt, die jeweils verfügbar war.

Da die Datenmenge, die die Wetterstationen liefern, erheblich ist und andererseits der eigentliche Zug nur einen sehr kleinen Teil des Beobachtungszeitraums einnimmt, wurde die Übernahme der Wetterdaten gleich zu Beginn auf die für den Zug relevanten Tage reduziert. Für diese Tage wurden die Werte je nach Verfügbarkeit stündlich, halbtäglich oder täglich übernommen.

3 Aufbereitung und Analyse im GIS

Sämtliche weitere Untersuchungen wurden im Geoinformationssystem ArcGIS pro durchgeführt. Dabei wurde ein Kartenausschnitt von Deutschland gewählt, der im NO Bornholm in Dänemark und im SW Troyes in Frankreich zeigt. Die Brutgebiete sind nördlich von Berlin in der Nähe von Parchim. Dieses Gebiet zählt zu den

größten Rastplätzen für Kraniche in Deutschland. Auf dem Weg zum Überwintern im Naturpark Forêt d'Orient und dem Lac du Der-Chantecoq gibt es ein weiteres Gebiet in dem sich die Kraniche gerne aufhalten. Dieses Gebiet ist die Diepholzer Moorniederung, welches im milden Winter oft als erstes Überwinterungsgebiet genutzt wird. Dazwischen liegt die im Wesentlichen über Westdeutschland und Frankreich führende Westeuropäische Zugroute.



Abbildung 3: typischer Routenverlauf

3.1 Visualisierung des Zugverhaltens

Um die Flugrouten als Ganzes darstellen zu können, wurden die isolierten GPS-Flugdatenpunkte zu einer Linie zusammengeführt und um Pfeile, die die Flugrichtung symbolisieren, ergänzt. Um den Standort der Kraniche darzustellen, werden Stecknadelsymbole genutzt (Abb. 3/4). In Abbildung 3 ist die Route 2014 zu

sehen. Es lassen sich gut die Aufenthaltsorte des getrackten Tieres bestimmen: Das Brutgebiet liegt in der Schweriner Umgebung, das Überwinterungsgebiet am Lac du Der-Chantecoq in Frankreich.

Durch Einbindung der jeweiligen Zeitstempel, war es darüber hinaus für die interaktive Analyse möglich, die Zeitachsenfunktion von ArcGIS zu benutzen und so auch die zeitliche Abfolge der dynamischen Bewegung von Wetter und Kranichen zugänglich zu machen. In Abbildung 4 sind alle zur Verfügung stehende GPS-Daten von 2013-2021 eingeflossen.

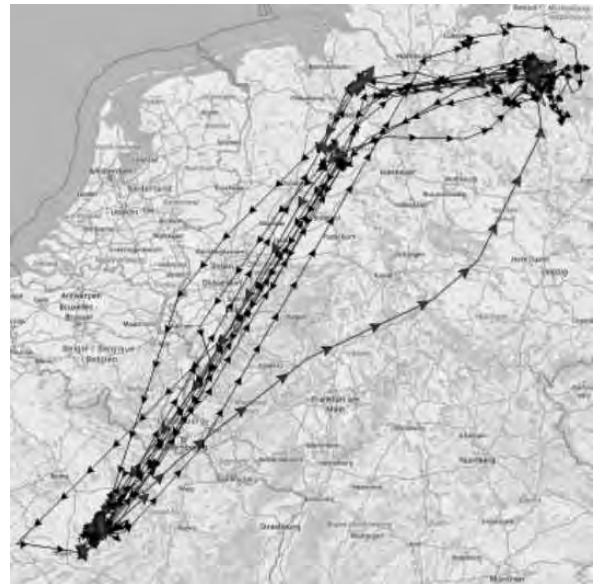


Abbildung 4: Routenverlauf 2013 bis 2021

Dabei zeigt sich bereits eine Auffälligkeit: Am 22.02.2016 kann man eine vom gewohnten Zugverhalten gänzlich abweichende Route beobachten (Abb.4, rot markiert).

3.2 Visualisierung der Wetterparameter

Den Wettermessdaten wurde zunächst ein Farbverlauf hinzugefügt, abhängig von den jeweiligen Messeinheiten, um die Temperatur, Niederschlag und Windstärke zu symbolisieren (Abb. 5 links). Aufgrund der verschiedenen Anzahl der Messstationen zwischen den einzelnen Messwerttypen und der besseren optischen Darstellung wurden mit Interpolations-Werkzeugen "Heatmaps" erzeugt. Abb. 5 mitte zeigt ein typisches Ergebnis, hier für den gemessenen Niederschlag.

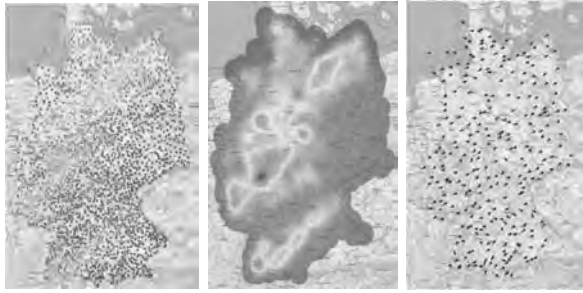


Abbildung 5: Temperaturdaten als Punktfeature (links); Niederschlagsdaten interpoliert als HeatMap (Mitte); Windrichtung mit Pfeilsymbolik (rechts)

Bei der Windrichtung hingegen wurden die Messstationen mit Pfeilen symbolisiert, die sich entsprechend der Windrichtung drehen (Abb. 5).

3.3 Zusammenhänge und Abhängigkeiten

Mit diesen Basisdarstellungen für Temperatur, Niederschlag und Windstärke sowie der eigentlichen Flugroute konnten anschließend mit Hilfe der Layertechnik des GIS beliebige Parameter übereinander dargestellt werden und mit der hier nicht darstellbaren Animationsfunktion des Zeitverlaufes sowohl räumlich als auch zeitlich in Beziehung gesetzt werden. Im Zentrum der Analyse stand dabei eine schlüssige Begründung für die „ungewöhnliche“ Route im Frühjahr 2016.

In Abbildung 6 ist sehr gut zu erkennen, dass sich die Flugroute entlang der Windrichtung anpasst. Am 22.02.2016 schien der Wind besonders günstig zu stehen und so flogen die Kraniche in Massen entlang des Luftstroms. Die aktuelle Position der abgebildeten Momentaufnahme ist als türkise Stecknadel auf der Abbildung zuerkennen.

In Abbildung 7 ist ein ähnliches Flugverhalten bei dem herbstlichen Flug in die Überwinterungsgebiete zu beobachten. Hier trafen sich die Kraniche erst einige Wochen zuvor im Diepholzer Moor, um sich zu sammeln und flogen dann mit dem günstigen Wind nach Frankreich.

4 Ergebnisse und Interpretation

Bei der Arbeit konnte festgestellt werden, dass die beobachteten Kraniche im Frühjahr 2016 eine andere Flugroute nutzen als in den anderen gemessenen Jahren. Betrachtet man Windstärke und -richtung, fällt auf, dass die Windrichtung entlang der von den Kranichen gewählten Route verläuft und die Windgeschwindigkeit höher ist (Abb. 8). In der Abbildung ist zu erkennen, dass die Route sich sehr eng an den Windrichtungspfeilen der Messstationen orientiert. Damit scheinen die besonderen Windverhältnisse des Frühjahrs 2016 Grund für die Abweichung von der gewohnten Flugroute gewesen zu sein.

Auch zeitliche Zusammenhänge zwischen den Windparametern Windstärke und Windrichtung und dem Zugverhalten lassen sich aus den Visualisierungen plausibel ableiten: Dabei ist es auffällig, dass die Massenzugtage tatsächlich dann beginnen, wenn der Wind günstig steht.

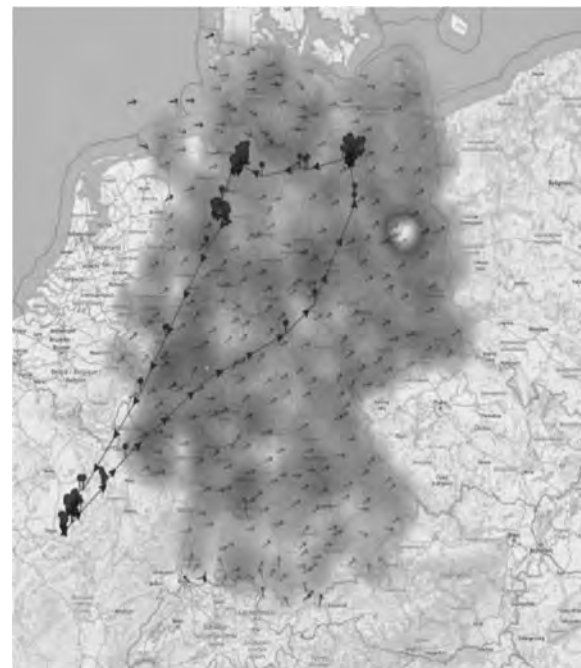


Abbildung 6: Temperatur, Windrichtung, 22.02.2016

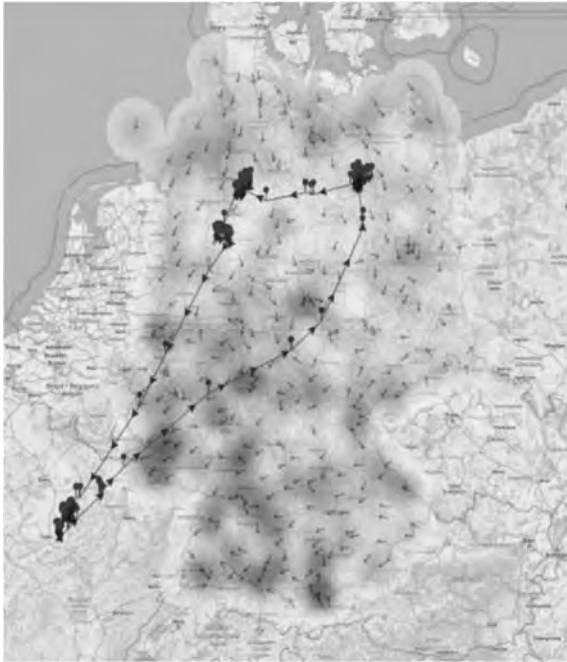


Abbildung 7: Windstärke, Windrichtung, 24.11.2016

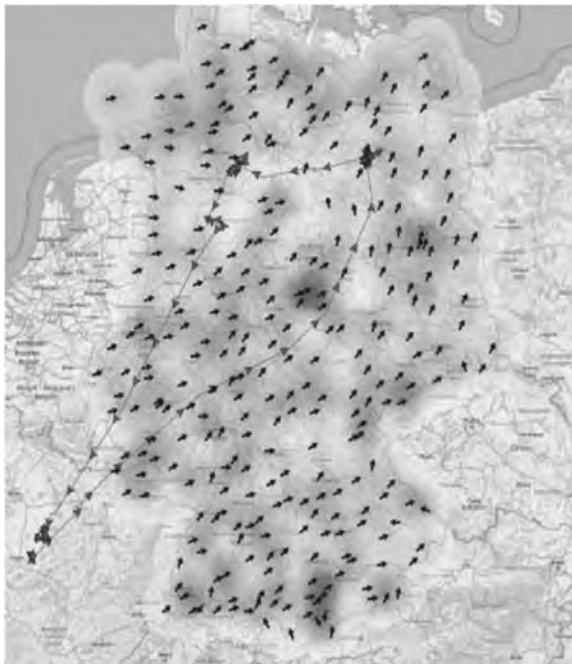


Abbildung 8: Kombination von Windrichtung, Windstärke und Route der Kraniche im Frühjahr 2016

Ein Zusammenhang der Flugrouten mit Temperatur und Niederschlag konnte dagegen nicht festgestellt werden.

5 Zusammenfassung

Das hier vorgestellte Pilotprojekt zeigt, dass die Analysefunktionen eines GIS eine raum-zeitliche Untersuchung der Trackingdaten ermöglichen. Bereits mit nur wenigen getrackten Tieren sind fundierte Aussagen über das Zugverhalten in Abhängigkeit von den Wetterbedingungen möglich. Mit Zugriff auf zusätzliche Kranichdaten und eine Ausweitung des Untersuchungsintervalls könnte nach Vorbild dieses Projekts der Zusammenhang von Wetter und der Routenplanung der Kraniche verstanden werden und es könnte in Zukunft möglich sein, mithilfe von meteorologischen Vorhersagen eine recht präzise Prognose zu stellen, wann die Kraniche ihre Langstrecken-Flüge beginnen und wo die Routen entlangführen könnten. Das Ausstatten von genügend Tieren ist jedoch sehr aufwändig und teuer, weshalb bislang (Stand Herbst 2021) nur vier Graukraniche in Deutschland besendert worden sind.

Aus technischer Sicht ist zu konstatieren, dass die hier aufgezeigten Ergebnisse ausschließlich aus interaktiver Arbeit mit der Bedienoberfläche des GIS gewonnen wurden. Wenn man die Untersuchungen ausweitet (sowohl zeitlich durch Einbeziehung weiterer Jahre als auch in Bezug auf die Anzahl der verfügbaren Tracks), erscheint es sinnvoll und erfolgsversprechend, die Analyse zu automatisieren. Hier besteht Potenzial bezüglich der Erstellung der Kartendarstellungen einerseits, andererseits aber auch in Bezug auf eine automatische Auswertung, die auf der Basis von neu zu entwickelnden Kennzahlen und einer entsprechenden KI-Komponente zur Bewertung der Übereinstimmung zwischen Wetterdaten und Zugbild basieren könnte.

Referenzen

- [1] Kranichschutz Deutschland, <https://kraniche.de/>
- [2] Climate-Data-Center, <https://cdc.dwd.de/portal/>
- [3] Die Welt der Kraniche, Hartwig Prange, 2016

Modelling Blood Volume Oscillations in the finger.

Hannah Kastinger^{1*}, Martin Bachler^{1, 2}

¹Institute for Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8-10,
1040 Vienna, Austria;

²Center for Health & Bioresources, AIT Austrian Institute of Technology, Giefinggasse 4,
1210 Vienna, Austria;

**hannah.kastinger@gmx.at*

Abstract. The aim of this work is to describe a model of the relationship between blood volume and transmural pressure of a vessel of the finger. The finger is pressed against a surface with increasing pressure. The focus lies on the amplitudes of the blood volume oscillations. The model leads to a formula for the blood volume oscillation amplitudes as a function of the contact pressure of the finger. Using model fitting, this equation can in the future be used for blood pressure estimation.

Introduction

This work introduces and describes a model of the relation between blood volume and transmural pressure of a vessel of the finger which is pressed against a surface. More of interest than the absolute volume are in fact the blood volume oscillations with respect to changes in the transmural pressure. This model for the blood volume oscillations shall be used for estimating the blood pressure without a cuff at the finger.

1 The Model

The idea of the model is to describe a relation between blood pressure inside a vessel, pressure acting on the vessel from outside and the blood volume inside the vessel. The vessel of interest, the transverse palmar arch artery, is located in the finger tip and the situation of interest is the fingertip being pressed on some surface with increasing pressure. This approach can be compared to the blood pressure measurement with a cuff around the upper arm as there is also an external pressure acting on the vessel. Similar to the finger pressing on a surface, the cuff is exerting decreasing pressure on the vessel from outside, while pressure oscillations are measured. First, let us take a look at the absolute volume of the vessel and afterwards at the blood volume

oscillations in dependency of the external pressure acting on the vessel from outside.

1.1 The Vessel Volume

The transverse palmar arch artery lies parallel to the surface of the finger and the bone underneath. Let us assume the artery as a cylindric tube of length L . When the finger presses on a surface, the pressure also acts onto the wall of the vessel. This external pressure (P_{ext}) is directed inwards. [1]

The second relevant component is the blood pressure (P_b). This is the pressure applied on the vessel wall by blood running through the arteries. It is directed opposite to P_{ext} , in direction of the outer normal vector. Hence, the pressure acting on the vessel wall is a combination of P_b and P_{ext} . This combination is defined as transmural pressure P_t as follows:

$$||P_t|| := ||P_b|| - ||P_{ext}||.$$

A positive transmural pressure P_t means that the blood pressure P_b exceeds the external pressure P_{ext} . This leads to the vessel being pressed apart. A negative transmural pressure P_t , on the other hand, means that P_{ext} exceeds P_b and therefore the vessel is compressed. [1]

To find a relation between the transmural pressure P_t and the blood volume V of the artery, A is defined as the cross-sectional area of the artery. First we are looking at the artery in relaxed state: $P_t = 0$. In this case, we assume A as a circular disc with radius $R \in \mathbb{R}$. The second case we are looking at is $P_t < 0$. In this case, the artery is compressed. Figure 1 shows the cross-section with side length $a \in \mathbb{R}^+$ and radius of the semicircles $0 \leq r \leq R$. [1]

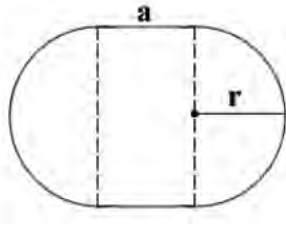


Figure 1: Schematic representation of the cross-section of the artery for a transmural pressure $P_t < 0$ [2].

The arterial wall is not stretched. We want to use this fact to find a relation between R , r and a . Let γ_0 be the boundary of A , the cross-section of the artery in relaxed state, then $L\gamma_0 = 2R\pi$. Let γ_s be the boundary of the deformed cross-section, then $L\gamma_s = 2r\pi + 2a$.

$$L\gamma_s = L\gamma_0 \quad (1)$$

$$2r\pi + 2a = 2R\pi \quad (2)$$

$$a = \pi(R - r) \quad (3)$$

$$\Rightarrow a \in [0, R\pi]. \quad (4)$$

If the artery is relaxed then $a = 0$ and if $a = R\pi$ the vessel is fully occluded. [1]

The next step is to find a relation between r and P_t . Let us define $r(t) := R(\sqrt{1 - e^{\alpha P_t}})$, $P_t < 0$, inspired by Charles. F. Babbs [3]. The parameter $\alpha > 0$ is related to the stiffness of the vessel wall. Now we can take a closer look at V .

$$V(P_t) = LA(P_t) = L(r(P_t)^2\pi + 2r(P_t)a) = \quad (5)$$

$$= LR^2\pi e^{\alpha P_t} = V_0 e^{\alpha P_t}, \quad (6)$$

where $V_0 := LR^2\pi$ is the volume of the relaxed vessel. We can see that a negative transmural pressure leads to an exponential reduction in blood volume. [1]

The next step is to look at the case $P_t > 0$. The wall is stretched uniformly in angular direction by the pressure. Hence, the cross-section A forms a circle with radius $r > R$. Let us define r in dependency of P_t based on Charles F. Babbs [3] as follows:

$$r(P_t) := R\sqrt{1 + \frac{\alpha}{\beta}(1 - e^{-\beta P_t})}. \quad (7)$$

The parameter $\beta > 0$ is also related to the stiffness of the arterial wall and is inversely proportional to the elastic-

ity of the vessel wall. Again we want to take a look at the volume V :

$$\begin{aligned} V(P_t) &= LA(P_t) = L(r(P_t)^2\pi = \\ &= LR^2\pi(1 + \frac{\alpha}{\beta}(1 - e^{-\beta P_t})) = \\ &= V_0(1 + \frac{\alpha}{\beta}(1 - e^{-\beta P_t})). \end{aligned} \quad (8)$$

[1]

All together we can define the vessel volume V as follows:

Definition 1.1 (Vessel volume; P. Baumann [1], p. 42)

Let $L > 0$ be the length of the cylindric shaped transverse palmar arch artery and $R > 0$ the radius of the cross-section in a relaxed state ($P_t = 0$). Furthermore, let $\alpha, \beta > 0$ be fixed parameters describing the vessel wall elasticity and assume the deformation behaviour described previously. Then, V is given by

$$V : \mathbb{R} \rightarrow \mathbb{R}, P_t \mapsto \begin{cases} V_0 e^{\alpha P_t}, & P_t < 0 \\ V_0(1 + \frac{\alpha}{\beta}(1 - e^{-\beta P_t})), & P_t \geq 0 \end{cases} \quad (9)$$

With this definition we can see that $V \in C^1(\mathbb{R})$ and that it is monotonically increasing. Figure 2 shows the function of the vessel volume V for estimated values for α and β . [1, 3]

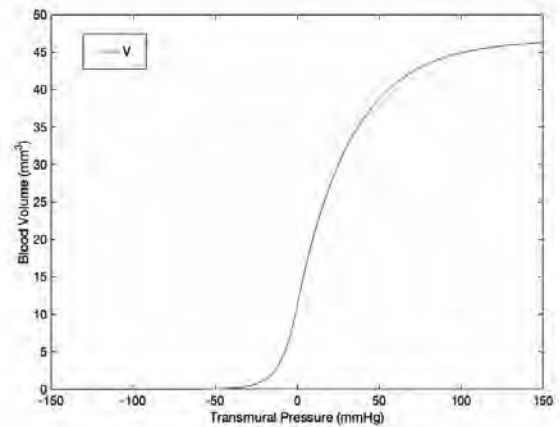


Figure 2: V for $\alpha = 0.11$ and $\beta = 0.03$.

1.2 The Oscillation Curve

The blood pressure P_b is not constant. In contrary, it varies between a minimum (diastolic, P_{dia}) and a maximum pressure (systolic, P_{sys}). If the external pressure increase at the finger is sufficiently slow, we can assume that P_{ext} is constant during each heartbeat. That means that the change of P_t is determined just by the variance of the blood pressure P_b . Furthermore, the differences of P_t lead to a varying amount of blood volume. Hence, V oscillates between its maximum and minimum during each cardiac cycle. Since V is monotonically increasing with P_t , it attains its minimum and maximum at the minimum and maximum of P_b . [1]

To make the next step let us first define pulse pressure $PP := P_{sys} - P_{dia}$. Now, we can define the maximum difference of blood volume ΔV during one heartbeat as follows:

$$\Delta V(P_m) := V(P_m + PP) - V(P_m). \quad (10)$$

P_m denotes the minimal transmural pressure which can also be calculated as $P_m = P_{dia} - P_{ext}$. [1]

The next step is to establish a connection between systolic and diastolic blood pressure and ΔV . First we will look at the case $P_m \leq -PP$:

$$\Delta V(P_m) = V_0 e^{\alpha(P_m + PP)} - V_0 e^{\alpha P_m} \quad (11)$$

The next case is $P_m \in (-PP, 0)$:

$$\Delta V(P_m) = V_0 \left(1 + \frac{\alpha}{\beta} (1 - e^{-\beta(P_m + PP)})\right) - V_0 e^{\alpha P_m} \quad (12)$$

Last, we will take a look at the case $0 \leq P_m$:

$$\Delta V(P_m) = \quad (13)$$

$$= V_0 \left(1 + \frac{\alpha}{\beta} (1 - e^{-\beta(P_m + PP)})\right) - V_0 \left(1 + \frac{\alpha}{\beta} (1 - e^{-\beta P_m})\right)$$

[1]

As already mentioned above, $P_m = P_{dia} - P_{ext}$. Therefore

$$\begin{aligned} P_m + PP &= P_{dia} - P_{ext} + P_{sys} - P_{dia} \\ &= P_{sys} - P_{ext} \end{aligned}$$

Finally, we can combine these results and formulate an

equation for ΔV :

$$\Delta V(P_{ext}) = \quad (14)$$

$$\begin{cases} V_0 \cdot e^{\alpha(P_{sys} - P_{ext})} - \alpha \cdot V_0 \cdot e^{\alpha(P_{dia} - P_{ext})}, & P_{ext} \geq P_{sys} \\ V_0 \left(1 + \frac{\alpha}{\beta} (1 - e^{-\beta(P_{sys} - P_{ext})})\right) - \alpha \cdot V_0 \cdot e^{\alpha(P_{dia} - P_{ext})}, & P_{dia} < P_{ext} < P_{sys} \\ V_0 \left(1 + \frac{\alpha}{\beta} (1 - e^{-\beta(P_{sys} - P_{ext})})\right) - V_0 \left(1 + \frac{\alpha}{\beta} \cdot e^{-\beta(P_{dia} - P_{ext})}\right), & P_{ext} \leq P_{dia}. \end{cases}$$

[1]

Figure 3 shows the oscillation amplitude curve ΔV for an assumed systolic and diastolic pressure of 120mmHg and 80 mmHg and $\alpha = 0.11$, $\beta = 0.03$. [3, 4]

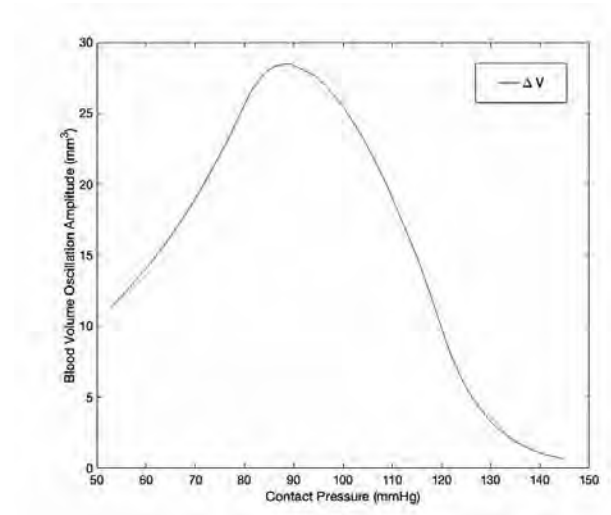


Figure 3: ΔV for $P_{sys} = 120\text{mmHg}$, $P_{dia} = 80\text{mmHg}$ and $\alpha = 0.11$, $\beta = 0.03$.

2 Conclusion and Outlook

This model, the formula for the blood volume oscillations ΔV in particular, in this form can be used for estimating blood pressure. To do so equation 14 can be used as model function and be fitted to blood volume oscillations measured from the transverse palmar arch artery. Among others, the systolic and diastolic blood

pressure are obtained as fitting parameters. A more detailed explication and evaluation of this approach will be part of future investigations and publications.

References

- [1] Philipp Baumann. *Model-based Method for cuff-less blood pressure measurement using the oscillometric finger-pressing method* [master thesis]. Institute for Analysis and Scientific Computing. Technische Universität Wien; 2020.
- [2] *Stadium Shape*
<https://www.calculatorsoup.com/calculators/geometry-plane/stadium.php>; accessed on 15.06.2022.
- [3] Charles. F. Babbs. *Oscillometric measurement of systolic and diastolic blood pressures validated in a physiologic mathematical model*. BioMedical Engineering OnLine. 2012; 11. doi: 10.1186/1475-925X-11-56.
- [4] Lachel Story. *Pathophysiology: a practical approach*. 3rd ed. Jones & Bartlett Learning; 2018.

SIR-type Epidemic Models for Education at Web

Alexander Freißlinger, Andreas Körner*, Felix Breiteneker

Institute for Analysis and Scientific Computing, TU Wien, Wiedner Hauptstrasse 8-10,
1040 Vienna, Austria; *Andreas.Koerner@tuwien.ac.at

Abstract. SIR-type epidemic models have become of big interest also for education, generally in modelling and simulation, and specifically in biomedical systems. As consequence, also various SIR-type models have been integrated into the MMT E-learning System used at TU Wien in various lectures on modelling and simulation. This contribution introduces the MMT E-Learning System and reports on the up to now implemented models and experiments for pandemic systems. The range starts with the basic Kermack-McKendrick SIR model with parameter studies and identification tasks, suggesting extensions to SIRS, SIRD, etc. models, - as case study for the simulation circle. Students are not only invited to 'play' with infection and contact parameters, but also to test lockdown strategies and vaccination strategies. Next, the classes or cohorts of exposed and quarantined people show extensions of the basic ODE SIR model dynamics, followed by MMT modules with adaptive lockdown strategies and with integration of mutated variants of a pathogen. Modules with sensitivity equations for the parameters discuss stability of the dynamics, and discrete model variants show the good coincidence of ODE models and difference equation models for the pandemic dynamics. And last but not least, an advanced MMT module introduces into spatial models for pandemics based on cellular automata.

Introduction

The SIR model is a compartment model which describes the relation between susceptibles (S), infected (I) and already recovered (R) individuals of a certain population during an epidemic.

It's origins date back to 1927: William Ogilvy Kermack und Anderson Gray McKendrick presented the original SIR model, in its initial form, a partial differential equation model that structures the infected population in terms of age-of-infection, while using simple compartments for people who are susceptible, infected, and recovered/removed. So the basic SIR model without age distinctions is given by the following ODEs:

$$\begin{aligned}\dot{S}(t) &= -i \cdot c \cdot S(t) \cdot I(t) \\ \dot{I}(t) &= +i \cdot c \cdot S(t) \cdot I(t) - r \cdot I(t) \\ \dot{R}(t) &= r \cdot I(t)\end{aligned}$$

Three model parameters characterize the dynamic behaviour, the infection-rate i , the contact-rate c , and the recovered-rate r ; furthermore, consistent initial values for the groups must be provided – and basis of the simplest SIR model in the MMT system.

The classical mathematical analysis investigates phase diagrams of S and I and their stability, based on the substitution of e.g. $R(t) = N - I(t) - S(t)$, N being the constant (normalized) total population number.

Analysis by numerical simulation allows the make use of time- and state-dependent parameters, which describe e.g. vaccination, change of the virus, etc., essential subjects in the experiments within the MMT system.

Unfortunately some infected people may die, adding a new term and the mortality rate d in the SIRD model:

$$\begin{aligned}\dot{S}(t) &= -i \cdot c \cdot S(t) \cdot I(t) \\ \dot{I}(t) &= +i \cdot c \cdot S(t) \cdot I(t) - d \cdot I(t) \\ \dot{R}(t) &= r \cdot I(t)\end{aligned}$$

Furthermore, re-infection of the recovered people may occur, adding another new term with re-infection parameter a in the SIRD model:

$$\begin{aligned}\dot{S}(t) &= -i \cdot c \cdot S(t) \cdot I(t) + a \cdot R(t) \\ \dot{I}(t) &= +i \cdot c \cdot S(t) \cdot I(t) - d \cdot I(t) - r \cdot I(t) \\ \dot{R}(t) &= r \cdot I(t) - a \cdot R(t)\end{aligned}$$

1 MMT E-Learning System

The MMT E-Learning System – MMT stands for Mathematics, Modelling and Tools – is a tool used at the Institute of Analysis and Scientific Computing and at TU Wien for education in modelling and simulation (and also by other institutes dealing with education in modelling and simulation).

The MMT server, developed since 2006, plays a major role in lectures for modelling and simulation and courses in applied mathematics at TU Wien. The case studies and examples for modelling and simulation deal with different kinds of modelling, like ODEs, cellular automata or agent-based models, and distinct applications. The MMT System is a web application with

- a frontend presenting interactive examples and case studies for modelling and simulation, as well as related lecture notes,
- with MATLAB running as simulation engine,
- and with a backend content management system for preparing examples, case studies etc., and lecture note content.



Figure 1: Entrance Webpage of the MMT Server.

When a student enters the website, he is welcomed by Adam Ries (1492–1559; a German mathematician), with login (Figure 1). After login for a certain course, the MMT system offers at left the modules of the course (Figure 2: *Pendulum Identification*, *Sim-Circle-Epidemics*, *LTI-Systems*, etc.).

When choosing the module *SIM-Circle-Epidemics*, the left menu extends with submodules, (Figure 2: modules *Basic SIR Model* to *SEIR Model*), and when finally choosing the module *Basic SIR Model* the experiment module pops up in the centre of the MMT window. It contains a description of the model, of the parameters, and of the experiment to be done. Finally the interface asks for parameter values, and with the OK-button the student starts the experiment, the SIR model simulation.

Now the MATLAB-engine of the MMT server executes the model file with ODE solver and ODE SIR model, which can accessed via the menu at the right (Figure 2: menu right, view m-file) – and gives back the results in the centre window below the experiment description (Figure 3).

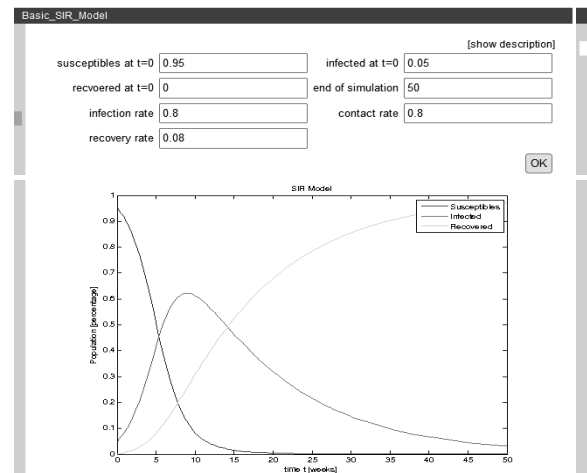


Figure 3: Result window of module *Basic SIR Model*.

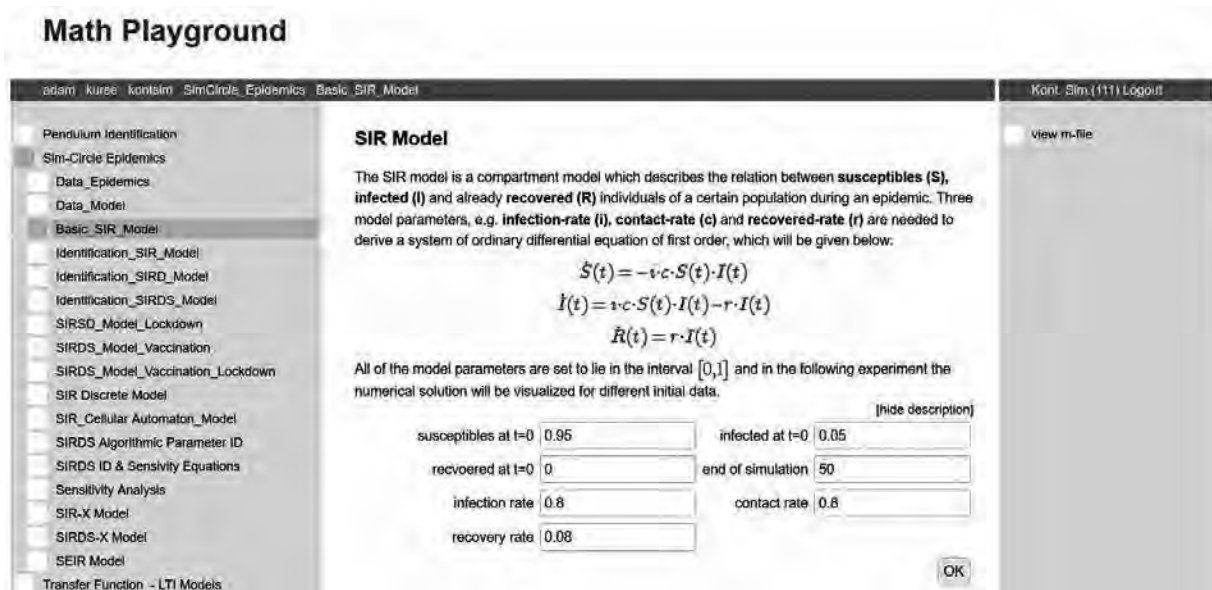


Figure 2: MMT Module Sim-Circle Epidemics, Experiment Basic SIR Model

2 MMT Simulation Circle Epidemics

The MMT server offers as first SIR module the simulation circle with the SIR case study, the module *Sim-Circle-Epidemics*. The opening page introduces into the simulation circle for identifying SIR-type models with traced data for infected and recovered people (Figure 4).

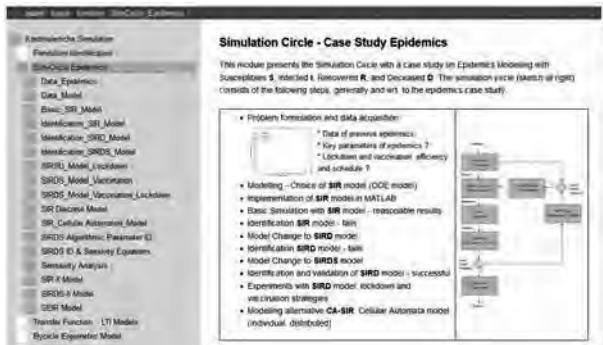


Figure 4: Intro for module *Sim-Circle Epidemics*.

The following modules *Data Epidemics* and *Data Model* offer to display the pandemic data, and to fit a polynomial model, in order to compare with a dynamic model. Then the module *Basic SIR Model* introduces into the classical SIR model as candidate for a proper model fitting the given pandemic data (Figure 2).

The module *Identification SIR Model* lets the student choose parameters – starting from defaults (Figure 5).

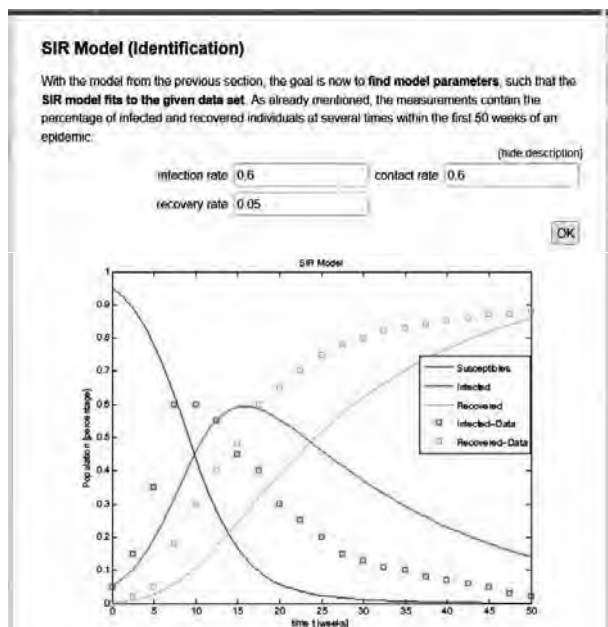


Figure 5: Default results *Identification SIR Model*.

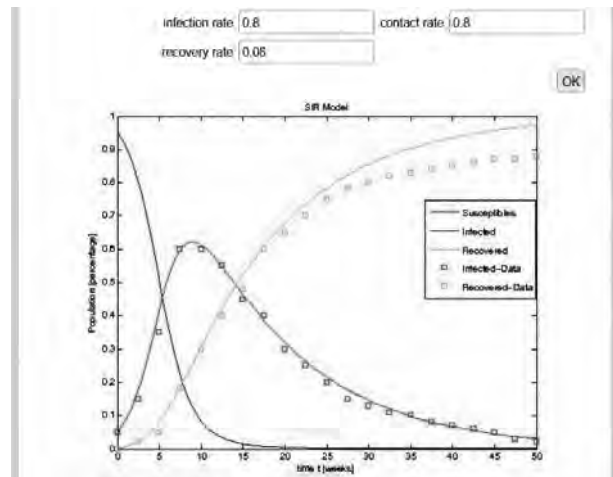


Figure 6: Optimal, but not satisfying results for module *Identification SIR Model*.

Students may now try better parameters, and clever students may have a look into the MATLAB model, where the optimal parameters are documented – but these are not really satisfying, as results in Figure 6 show.

As next step, the student may try more appropriate models, as the SIRD model, or the SIRDs model (sketched before, and documented also in the modules). Finally he will end up with the module *Identification SIRDs Model* – giving satisfying results for identification using heuristically the try-and-error strategy (Figure 7).

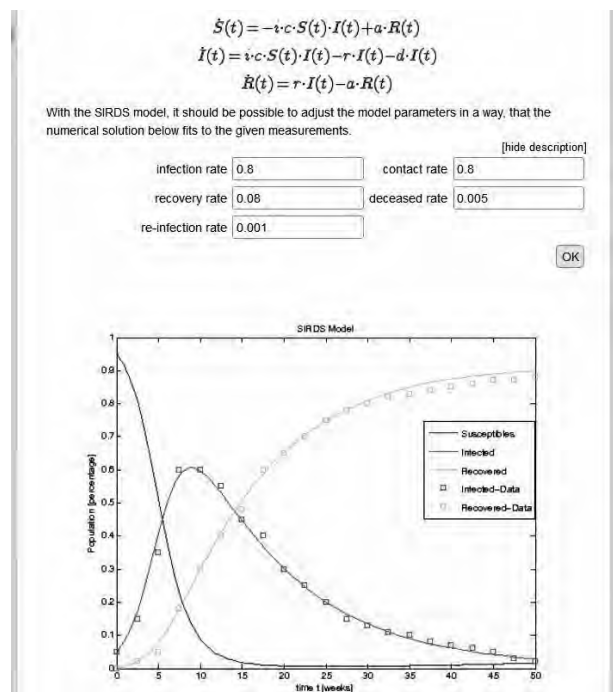


Figure 7: Optimal heuristic results for module *Identification SIRDs Model*.

Now the students are invited to experiment with lockdown strategies and vaccination strategies, implemented in modules *SIRSD Model Lockdown* and *SIRSD Model Vaccination* by time-dependent contact parameters and infection parameters. And finally, in the module *SIRSD Model Vaccination Lockdown* the student may act as minister for health care, simulating decisions for lockdown directives and vaccination campaigns (Figure 8).

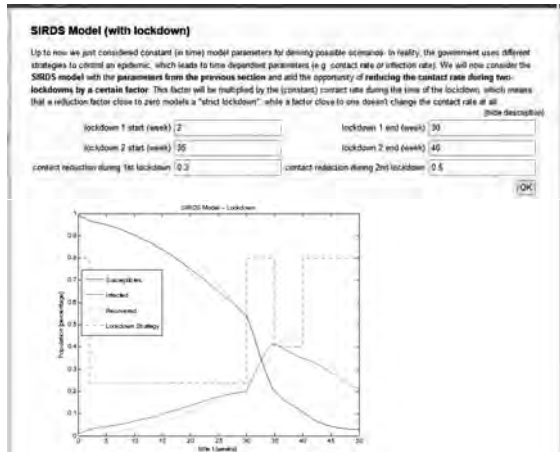


Figure 8: Lockdown and vaccination experiments in module *SIRSD Model Vaccination Lockdown*.

The following modules address advanced students. The modules *SIRSD Algorithmic Parameter ID* and *SIRSD ID & Sensitivity Equations* replace the heuristic parameter identification by introduction of the least square method, with numerical partial derivatives and with partial derivatives via sensitivity ODEs. Figure 9 gives a sketch of the documentation and of the (numeric) results.

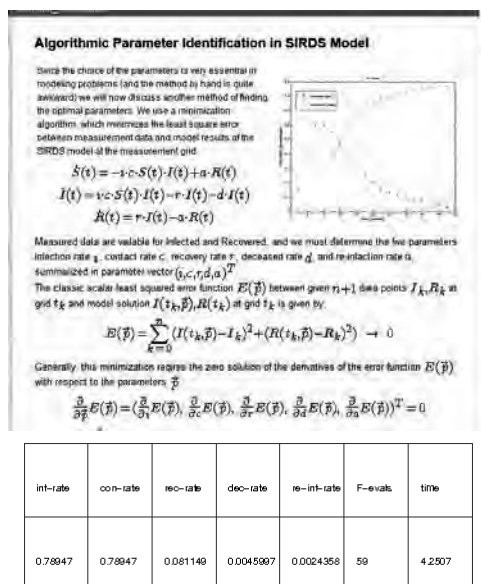


Figure 9: SIRSD model with least-square parameter identification.

3 Advanced SIR-type Modules

The MMT server offers also advanced SIR modules, as experiments with the SIR-X model and SIRDS-X, which integrates the group X of people in quarantine.

Of importance are SEIR models, which distinguish between exposed people E (infected, but not infectious), and the really infected people I – the MMT server provides her the basic SEIR model, as well as advanced models integration a virus mutation. There, the groups of exposed and infected people are doubled, and the equations are extended by a competitive behaviour of the virus types (Figure 9).

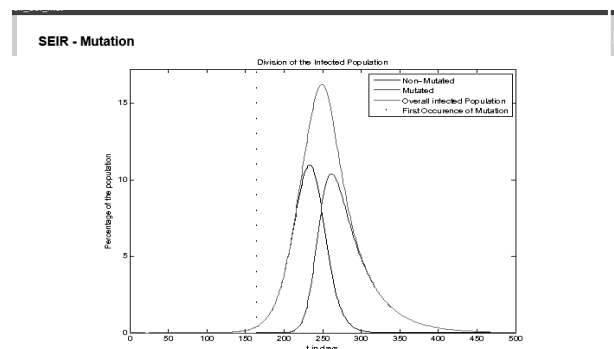


Figure 9: SEIR model with virus mutation.

And last but not last, the MMT server presents an almost ‘exotic’ approach to SIR dynamics, including spatial behaviour – the module *SIR Cellular Automata*. Susceptible (blue), vaccinated (light blue), infected (red), and recovered (green) people move on a spatial grid, infect neighbours, and undergo the phases of epidemics by means of contact rules and phase change rules. So on the grid the pattern of colours represents a snapshot of virus distribution (Figure 10, above) – but summing up the people to groups, again the typical SIR dynamics appear (Figure 10, below).

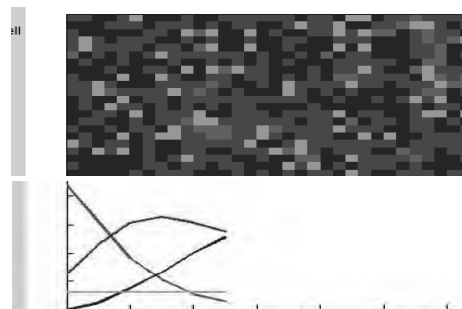


Figure 10: Results module *SIR Cellular Automata*.

And there are some more SIR modules: *SIR Discrete Model*, *SIR Random Rates*, *SIR Sensitivity*, ...

Simulationsmethoden

Methods in Simulation

A time domain approach for system identification using hill climbing (L)
R. Büchi

Geometrisches Mehrgitterverfahren für die gyrokinetische Poisson-Gleichung mit
Anwendung auf Kernfusion (K)
C. Schwarz, P. Leleux, M. J. Kühn, U. Rüde

Using the Proxel Method to build EHMM for behaviour reconstruction (L)
P. Krenckel, C. Krull

An Extension for the Specification and Automated Selection of System Variants Based
on the System Entity Structure Using a Problem from Process Industry (L)
H. Folkerts, T. Pawletta, U. Durak

Hestia jl: A Julia library for heat conduction modeling with boundary actuation (K)
S. Scholz, L. Berger

Analysis of Selected Literature on Co-simulation (L)
I. Hafner, S. Emrich, N. Popper

Coupling of Simulation Tools for Obtaining Local Fatigue in Combination with
Experimental Data (K)
N. Adamchuk, B. Rösch, W. Schlüter, F.-C. Reissner, J. Baumgartner

Hochaufgelöste Energieprofile durch hybride Simulation (K)
B. Woerrlein, S. Strassburger

Modular platform for route guidance in the cyber-physical laboratory test field (L)
M. Göllner, S. Jacobitz, T. Li, X. Liu-Henke

Untersuchung der anwendungsspezifischen Verifikations- und Validierungstechniken
unter Berücksichtigung einer veränderten Datenbasis (K)
K. Langenbach, A. A. Scheidler, M. Rabe

Service-oriented Architecture for Automatic Evaluation of Urban Logistical Concepts (L)
R. Pump, L. Bienzeisler, T. Lelke, V. Ahlers, A. Koschel

Die Langbeiträge (L) sind zu finden im **ARGESIM Report 20,**
Proceedings Langbeiträge ASIM SST 2022

ISBN ebook 978-3-901608-97-1, DOI: 10.11128/arep.20, ASIM Mitteilung AM 180

Geometrisches Mehrgitterverfahren für die gyrokinetische Poisson-Gleichung mit Anwendung auf Kernfusion

Christina Schwarz¹, Philippe Leleux², Martin J. Kühn³, Ulrich Rüde⁴

¹Lehrstuhl für Wissenschaftliches Rechnen, Universität Bayreuth

²Cerfacs, Toulouse

³DLR, Inst. für Softwaretechnologie, Department für High-Performance Computing

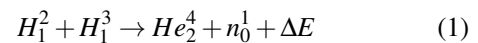
⁴Lehrstuhl für Systemsimulation, FAU Erlangen-Nürnberg

Abstract. Um den Klimawandel zu bekämpfen und unser Ökosystem zu erhalten, müssen wir den Ausstoß von Kohlendioxid in die Atmosphäre reduzieren. Eine vielversprechende Alternative zu erneuerbaren Energien ist die Kernfusion, welche eine nahezu unendliche Menge an sauberer und sicherer Energie liefern könnte. Die Reaktion ist jedoch extrem komplex und instabil und kann daher noch nicht lange genug aufrecht erhalten werden. Da der Bau und Betrieb von Fusionsreaktoren außerordentlich teuer ist, sind numerische Simulationen erforderlich, um unser Wissen über den Fusionsprozess zu erweitern. Der existierende Code *GyselaX* für Plasmasimulationen löst als Teilproblem eine zweidimensionale Poisson-Gleichung auf dem Querschnitt der Reaktor-geometrie. Das EU-Projekt EoCoE zielt auf die Verbesserung des derzeitigen Löser für diese Gleichung ab, um die Simulationszeiten zu reduzieren. Hierzu wurden ein geometrischer Mehrgitteransatz unter Verwendung Finiter Differenzen für die Diskretisierung, ein kombiniertes Verfahren zur Linienglättung und eine implizite Extrapolationstechnik entwickelt [1, 2], implementiert und optimiert.

1 Kernfusion zur Energiegewinnung

Eine vielversprechende Alternative zu erneuerbaren Energien, um den Ausstoß von Kohlenstoffdioxid bei der Energiegewinnung zu reduzieren, ist die Kernfusion [3, 4]. Diese auch von der Sonne genutzte Umwandlung von Masse in Energie nach dem Einsteinschen Gesetz $E = mc^2$ ist die einzige bisher auf Erden noch un-

genutzte Energiequelle. Bei der Reaktion



verschmelzen leichte Wasserstoff-Isotope (Deuterium und Tritium) zu schwereren Helium-Kernen (α -Teilchen), wobei Energie freigesetzt wird. Der Vorteil im Vergleich zur bekannten und umstrittenen Kernspaltung, bei der Energie durch die Spaltung von schwerem Uran in leichtere Atome frei wird, sind die Freisetzung der ca. 50-fachen Energiemenge und die weniger gefährlichen radioaktiven Reaktionsprodukte. Mit reichlich Ressourcen auf der Erde für die Reaktion hat die Kernfusion ein enormes Potential als CO₂ neutrale Energiequelle der Zukunft. Allerdings ist die Reaktion höchst instabil und nur sehr schwer aufrecht zu erhalten, sodass die Energiebilanz zum derzeitigen Forschungsstand noch viel zu gering ist, um kommerziell genutzt zu werden.

Da für den Fusionsprozess auf Erden sehr hohe Temperaturen benötigt werden, bei denen das Gas als sog. *Plasma* vorliegt, ist ein Einschluss in einer torus-förmigen Vakuumkammer mit Hilfe von starken Magnetfeldern nötig. In dem sogenannten *Tokamak*-Reaktor, wird durch äußere Spulen ein toroidales Magnetfeld erzeugt, entlang dessen Feldlinien sich die geladenen Atomkerne und Elektronen kreisend bewegen. Um das Abdriften von Teilchen von ihrer Kreisbahn zu vermeiden, wird ein zusätzliches, durch einen induzierten Plasmastrom im Inneren des Reaktors erzeugtes, poloidales Magnetfeld überlagert. Insgesamt ergibt sich somit ein helix-förmiges Magnetfeld, siehe Abb. 1.

Da die Reaktionen in einem Tokamak sehr komplex und bis heute noch nicht komplett erforscht sind, ist es jedoch noch ein weiter Weg bis zur tatsächlichen Nut-

zung von Fusionsenergie. Um unser Wissen über Plasmafusion zu erweitern, werden zunächst Simulationen benötigt, nachdem die Kosten für Fusionsreaktoren und Experimente extrem hoch sind.

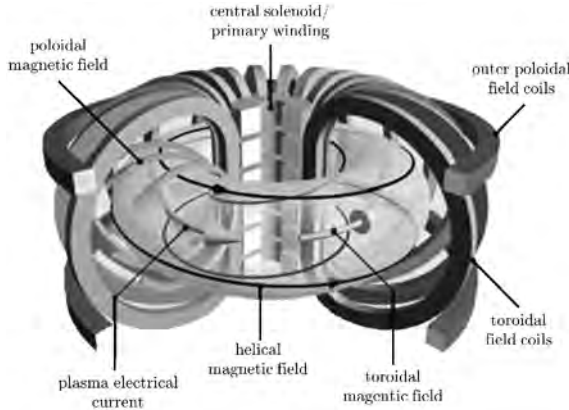


Figure 1: Tokamak Konzept mit helix-förmigen Magnetfeldlinien, abgebildet mit Erlaubnis von Eurofusion.

2 Plasmasimulation mit GyselaX

Auch die Simulation von Plasma ist mit hohen Kosten in Bezug auf Rechenzeit und Speicherkapazität verbunden und macht daher die Entwicklung von effizienten Codes nötig. Von besonderem Interesse sind in diesem Zusammenhang gyrokinetische Codes [5, 6], die in der Lage sind, Turbulenzen im Plasma zu erfassen, und es ermöglichen, die Dimension des Systems von 6 (3 für die Torusgeometrie und 3 für die Geschwindigkeit) auf 5 zu reduzieren. In aktuellen Implementierungen, wie *GyselaX* (GYrokinetic SEmi-LAgrangian code) [5], muss in jedem Zeitschritt eine 5D Vlasov-Gleichung gelöst werden, die mit einer 3D Poisson-Gleichung gekoppelt ist. Letztgenannte wird nochmals auf die Lösung einer 2D Poisson-Gleichung auf mehreren poloidalen Querschnitten des Reaktors reduziert.

Die Lösung dieses 2D-Systems ist zwar von geringerer Dimension, aber immer noch sehr rechenintensiv. Aus diesem Grund ist eines der Ziele des EU-Projektes *EoCoE: Energy oriented Center of Excellence: towards exascale for energy* die Beschleunigung des Löser dieser 2D Poisson-Gleichung. Zu diesem Zweck wurde ein *geometrisches Mehrgitterverfahren in verallgemeinerten Polarkoordinaten mit impliziter Extrapolation*

zusammen von der FAU Erlangen-Nürnberg und dem CERFACS in Toulouse entwickelt, welches im Folgenden vorgestellt wird.

3 Geometrisches Mehrgitterverfahren in verallgemeinerten Polarkoordinaten für eine 2D Poisson-Gleichung mit impliziter Extrapolation

Die gegebene partielle Differentialgleichung ist auf dem Querschnitt des Tokamak-Reaktors definiert als

$$\begin{aligned} -\nabla \cdot (\alpha \nabla u) &= f & \text{in } \Omega \\ u &= 0 & \text{on } \partial\Omega, \end{aligned} \quad (2)$$

wobei α der Diffusionskoeffizient ist, welcher eine Anisotropie in das zu lösende Problem miteinbringt, siehe [2].

Da die Gleichung auf dem kreisförmigen Querschnitt gelöst werden muss, eignen sich krummlinige Koordinaten (r, Θ) am besten zur Darstellung dieser Geometrie, siehe Abb. 2.

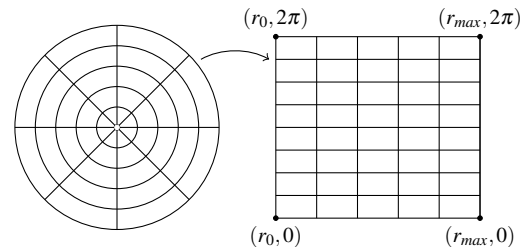


Figure 2: Abbildung des Gitters von kartesischen auf Polarkoordinaten

Insgesamt werden drei verschiedene vereinfachte Geometrien betrachtet [2], siehe Abb. 3.

Auf Grund des anisotropen Operators und des durch die Koordinatentransformation ungleichförmigen Gitters, können Standard Finite Differenzen-Ansätze zu unsymmetrischen Matrizen führen. In [1] wurde daher ein symmetrischer 9-Punkte Stencil durch Finite Differenzen auf Basis der Minimierung des Energiefunktions hergeleitet.

Das diskretisierte Problem $Au = f$ soll nun mit der Hilfe von Mehrgitterverfahren gelöst werden. Mehrgit-

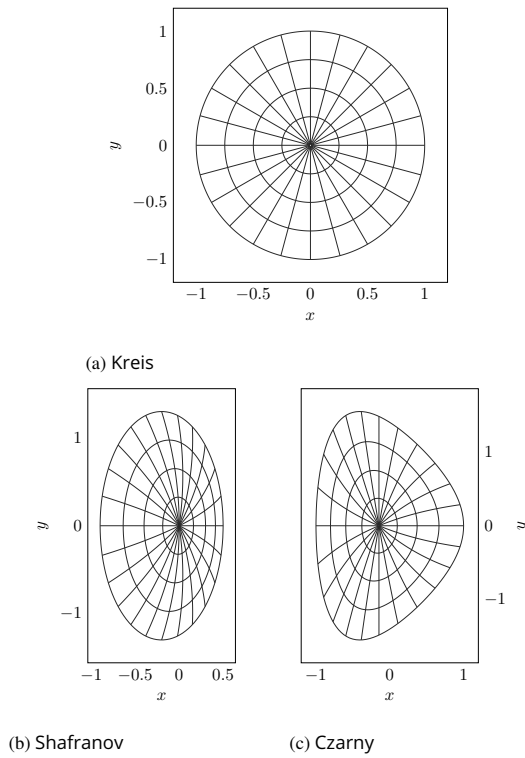


Figure 3: Verschiedene Geometrien für den poloidalen Tokamak-Querschnitt.

terverfahren sind schnelle, numerische, iterative Methoden zum Lösen von partiellen Differentialgleichungen, bei denen mehrere Gitterebenen mit unterschiedlich feiner Diskretisierung rekursiv kombiniert werden. Die Idee ist die Approximation des Fehlers auf den gröberen Gittern, bzw. das einfache exakte Lösen der Residuums Gleichung $Ae = r$ auf dem größten Gitter zur Verbesserung der Lösung auf den feineren Gittern.

Auf Grund der Verwendung von Polar- bzw. krummlinigen Koordinaten und einer zusätzlichen Verfeinerung des Gitters angesichts des Faktors α erhalten wir ein ungleichförmiges Diskretisierungsgitter und einen anisotropen Operator. Da punktweise Relaxation nur in Bezug auf stark gekoppelte Freiheitsgrade im Operator eine starke Wirkung zeigt und bei schwach gekoppelten Freiheitsgraden unzureichend ist, verwenden wir Linienglätter [7]. Hierbei wird gleichzeitig nach einer ganzen Gitterlinie an Unbekannten gelöst, wodurch sich ein lineares Gleichungssystem ergibt. Speziell beim Zebra-Linienglätter behandelt der Löser jeweils jede zweite Gitterlinie gleichzeitig, da in unserem Fall hier keine Abhängigkeiten bestehen [7]. Auf einem kreisförmigen Gebiet können zwei natürliche Lini-

englättungsoperationen unterschieden werden: die Aufteilung des Gitters in kreisförmige bzw. radiale Gitterlinien [2], siehe Abb. 4a und 4b. Da beide Verfahren je nach Position innerhalb des Gebiets unterschiedlich gute Ergebnisse liefern [8], wird hier ein kombiniertes Verfahren angewandt. Um eine gute Konvergenz zu erhalten, wird das Gebiet in zwei Abschnitte unterteilt und in der Nähe des Ursprungs kreisförmige, bzw. am äußeren Rand radiale Linien verwendet [2], siehe Abb. 4c.

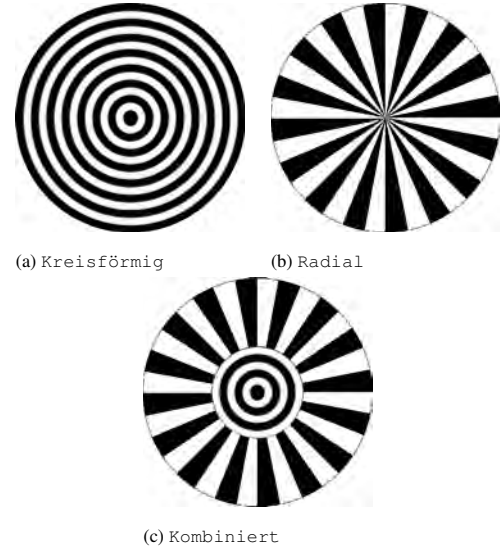


Figure 4: Verschiedene Zebra-Linienglätter.

Um die Genauigkeit des Löser noch zu verbessern, wird ein implizites Extrapolationsverfahren in das Mehrgitterverfahren integriert. Basierend auf der τ -Extrapolation [11] werden auf dem feinsten Gitter (F) zusätzlich Informationen des nächst größeren Gitters (C) verwendet, um den lokalen Fehler abzuschätzen und bestimmte, dominierende Fehlerterme zu eliminieren [9, 10]. Durch geeignete Kombination von Gleichungssystemen niedrigerer Ordnung

$$\begin{aligned} A^{extrapol} &= \begin{pmatrix} 4/3A_{F,cc} - 1/3A_C & 4/3A_{F,cf} \\ 4/3A_{F,fc} & 4/3A_{F,ff} \end{pmatrix}, \\ f^{extrapol} &= \begin{pmatrix} 4/3f_{F,c} - 1/3f_C \\ 4/3f_{F,f} \end{pmatrix}, \end{aligned} \quad (3)$$

kann implizit eine Lösungen höherer Ordnung erhalten werden [12].

Im Zusammenhang mit finiten Elementen ist eine Diskretisierung auf Basis linearer Elemente in Kombination mit impliziter Extrapolation gleichbedeutend mit

einer Diskretisierung mit quadratischer Elementen [10]. Besonders bei diesem Verfahren ist der Ausschluss der Gitterpunkte des groben Gitters bei der Glättung auf dem feinsten Gitter.

Numerische Ergebnisse aus [2] zeigen, dass dieses implizite und kostengünstige Extrapolationsverfahren die Konvergenzordnung von zwei auf vier erhöht.

4 Implementierung und Optimierung

Ziel war die Übersetzung des existierenden Matlab Codes in C++ zur Steigerung der Effizienz und Ermöglichung der Parallelisierung. Der Fokus lag auf einer Matrix-freien Implementierung, sodass große Operatoren nicht gespeichert, sondern direkt auf Vektoren angewandt und kleinere Operatoren im Sparse-Matrix-Format gespeichert werden. Durch die Reduktion von redundanten Operationen wurde die Effizienz des Codes noch weiter gesteigert. Außerdem wurde der Code mit OpenMP, sowie mit CUDA parallelisiert.

Weitere Optimierungsschritte, sowie numerische Ergebnisse werden im Vortrag präsentiert.

5 Ausblick

Der finale C++ Code liefert die selben Ergebnisse und Konvergenzen wie der zu Grunde liegende Matlab Code. Durch die Matrix-freie Implementierung, Optimierung und Parallelisierung konnte die Effizienz stark gesteigert werden. Zukünftig soll der Code noch mit anderen Ansätzen in Bezug auf Konvergenz, Genauigkeit und Kosten verglichen werden und schließlich in den Gysela-Code integriert werden, um die Laufzeiten und Gesamtkosten von Plasmasimulationen zu verringern.

Acknowledgement

Diese Arbeit wurde durch das European Union's Horizon 2020 Programm gefördert.

References

- [1] Kühn, M. J.; Kruse, C.; Rüd e, U. *Energy-Minimizing, Symmetric Discretizations for Anisotropic Meshes and Energy Functional Extrapolation*, SIAM J. Sci. Comput. Vol. 43(4), pp. A2448-A2473, 2021.
- [2] Kühn, M. J.; Kruse, C.; Rüd e, U. *Implicitly Extrapolated Geometric Multigrid on Disk-Like Domains for the Gyrokinetic Poisson Equation from Fusion Plasma Applications*, SIAM Journal of Scientific Computing, 91(1), 2022.
- [3] Stacey, W. M. *Fusion Plasma Physics*, Wiley-VCH, Weinheim, 2005.
- [4] Wesson, J. *Tokamaks*, Clarendon Press, Oxford, 1997.
- [5] Bouzat, N.; Bressan, C.; Grandgirard, V. et al. *Targeting Realistic Geometry in Tokamak Code Gysela*, ESAIM, 63, pp. 179-207 2018.
- [6] Grandgirard, V. et al. *A 5d gyrokinetic full-f global semi-Lagrangian code for flux-driven ion turbulence simulations*, Computer Physics Communications, 207, pp. 35-68, 2016.
- [7] Trottenberg, U.; Oosterlee, C.; Schüller, A. *Multigrid*, chapter 5, Acad. Press, San Diego, 2001.
- [8] Barros, S. *The Poisson equation on the unit disk: a multigrid solver using polar coordinates*, Applied Mathematics and Computation, 25(2), pp. 123-135, 1988.
- [9] Rüd e, U. *Extrapolation and Related Techniques for Solving Elliptic Equations*, TU München, Institut für Informatik, 1991.
- [10] Rüd e, U. *The Hierarchical Basis Extrapolation Method*, SIAM Journal of Scientific Computing, 13(1), pp. 307-318, 1992.
- [11] Rüd e, U. *Multiple tau-Extrapolation for Multigrid Methods*, TU München, Institut für Informatik, 1987.
- [12] Rüd e, U. *Extrapolation Techniques For Constructing Higher Order Finite Element Memethods*, TU München, Institut für Informatik, 1998.

Hestia.jl: A Julia library for heat conduction modeling with boundary actuation

Stephan Scholz*, Lothar Berger

Control and Process Engineering, Ravensburg-Weingarten University of Applied Sciences, Germany

*stephan.scholz@rwu.de

Abstract. Heat conduction modeling in three dimensions with boundary actuation plays an important role in thermal process engineering, for example in case of heating plates, laser welding or 3D printing. Here, the actuators and the induced energy have to be described exactly for such processes to guarantee a high simulation quality. We introduce Hestia.jl, a software library to model three-dimensional heat conduction with multiple spatially distributed heat sources on the boundary.

Introduction

The heat equation is a standard example in numerical analysis and control theory. Several software tools and libraries like OPENFOAM [1], FENICS [2], TRIxi.JL [3] and VORONoiFVM.JL [4] exist to solve heat equation models in one, two or three dimensions. These software tools are general purpose solvers for (specific types of) partial differential equations, which means they are also applicable for other models like the Poisson or Burgers' equation. However, they often require solid knowledge in the theory of finite volume or finite element methods.

We present HESTIA.JL [5], a Julia library [6] to simulate heat conduction in one, two and three dimensions with boundary actuation. A discretized heat conduction model is created in few steps using HESTIA.JL, without deep knowledge of numerical analysis. This heat conduction model is solved in time using high-order numerical integrators provided by DIFFERENTIALEQUATIONS.JL [7].

1 Problem formulation

Heat conduction is often modeled to occur inside geometrical objects like one-dimensional rods $\Omega = (0, L)$, two-dimensional plates $\Omega = (0, L) \times (0, W)$ and three-dimensional cuboids $\Omega = (0, L) \times (0, W) \times (0, H)$. These objects are implemented in HESTIA.JL as data

types HeatRod, HeatPlate and HeatCuboid to store the original dimensions and the spatial approximation including sampling and number of grid points.

We distinguish linear and quasi-linear heat conduction depending on the definition of its physical properties: thermal conductivity λ , specific heat capacity c and mass density ρ . Constant properties (in case of linear heat conduction) are stored as a StaticIsoProperty whereas temperature-dependent properties (in case of quasi-linear heat conduction) are stored as a DynamicIsoProperty. In particular, temperature-dependent properties are assumed to be modeled as power series, e.g. $\lambda(\Theta) = \sum_{n=1}^N a_n \Theta^{n-1}$, and the coefficients are saved in an array, e.g. $[a_1, \dots, a_N]$. So, we assume the quasi-linear heat conduction model

$$\rho(\vartheta) c(\vartheta) \frac{\partial \vartheta(t, x)}{\partial t} = \text{div} [\lambda(\vartheta) \nabla \vartheta(t, x)]$$

with $(t, x) \in (0, T) \times \Omega$ and final time $T > 0$ to be a generalization of the linear heat equation.

On boundary $\partial\Omega$ the variation of temperature ϑ is affected by the boundary conditions, namely linear heat transfer

$$-h(x) ([\vartheta(\cdot, x) - \Theta_{amb}(x)])$$

and nonlinear heat radiation

$$-\varepsilon(x) \sigma [\vartheta(\cdot, x)^4 - \Theta_{amb}(x)^4]$$

to the environment with ambient temperature Θ_{amb} , heat transfer coefficient h , emissivity ε and Stefan-Boltzmann constant σ , see also [8]. The parameters h , ε and Θ_{amb} are stored in data type Emission and can be defined for each boundary side separately, see Table 1.

West	$\{0\} \times [0, W] \times [0, H]$
East	$\{L\} \times [0, W] \times [0, H]$
South	$[0, L] \times \{0\} \times [0, H]$
North	$[0, L] \times \{W\} \times [0, H]$
Underside	$[0, L] \times [0, W] \times \{0\}$
Topside	$[0, L] \times [0, W] \times \{H\}$

Table 1: Names and positions of boundary sides.

2 Actuator configuration

Actuators are only assumed on boundary sides - not inside the geometrical object. The actuated boundary sides (west, east, etc.) are partitioned using a checker-board pattern and for each partition β_n an actuator and its spatial configuration b_p can be defined as introduced in [9]. This spatial configuration describes the possible spatially distributed intensity of actuation ranging from zero (no actuation) to one (full actuation). We define a radial symmetric configuration as

$$b_p(x) = \begin{cases} m_p \cdot \exp(-\|M_p(x - x_{c,n})\|^{2v_p}) & \text{for } x \in \beta_n, \\ 0 & \text{for } x \in B_A \setminus \beta_n \end{cases}$$

with scaling $m \in [0, 1]$, curvature matrix $M \in \mathbb{R}^{3 \times 3}$, power $v \in \mathbb{N}_{>0}$ and central point $x_{c,n} \in \beta_n$ of the n -th partition. These coefficients are stored as `RadialConfiguration`. The choice of a spatial configuration shall approximate real-world scenarios where heating elements may not be able to induce the same amount of energy at each position of its surface. An example partition and its related actuator configuration are illustrated in Figure 1.

3 Demonstration example

Next, we explain how to build a simulation of a cooling-down and heating-up process for a three-dimensional cuboid. A full listing can be found on GitHub [10].

Setting up the model

We specify the physical properties of a quasi-linear heat conduction model as $\rho := 7800$,

$$\begin{aligned} c(\Theta) &:= 330 + 0.4 \Theta \quad \text{and} \\ \lambda(\Theta) &:= 10 + 0.2 \Theta - 10^{-4} \Theta^2. \end{aligned}$$

These coefficients are stored in arrays as noted in Section 1 and a `DynamicIsoProperty` is created as in Listing 1.

```
property =
    createDynamicIsoProperty(
        [10.0, 0.2, -1e-4], [7800], [330, 0.4])
```

Listing 1: Create property type for quasi-linear dynamics.

We design a cuboid of length $L = 0.3$, width $W = 0.2$ and height $H = 0.1$, which is discretized by 40 cells in x_1 -, 24 cells in x_2 - and 10 cells in x_3 -direction. The model is built as in Listing 2

```
cuboid =
    HeatCuboid(0.3, 0.2, 0.1, 40, 24, 10, property)
```

Listing 2: Create cuboid model.

The ambient temperature of the boundary conditions is set to 300 Kelvin. On boundaries `west` and `east` we assume the heat transfer coefficient $h = 10$ and emissivity $\varepsilon = 0.6$, see Listing 3. On boundaries `north` and `east` we consider only heat transfer with $h = 10$ and no heat radiation, all other boundary sides are considered with zero-Neumann boundary conditions ($h = 0$ and $\varepsilon = 0$) and do not have to be implemented explicitly.

```
boundary = initBoundary(cuboid)
emission = createEmission(10, 0.6, 300)
setEmission!(boundary, emission, :west)
```

Listing 3: Specify boundary conditions.

Cooling-down process

Now, the heat equation is approximated in space and forms an ordinary differential equation (ODE) which is solved with `DIFFERENTIALEQUATIONS.JL`. The specification (geometry, property, boundary conditions) and the temperatures are handed over to the `diffusion!` function to compute the right-hand side of the ODE, see Listing 4. The `cool_down!` function forms a standard interface for numerical ODE integration methods provided by `DIFFERENTIALEQUATIONS.JL`.

```
cool_down!(dv, v, p, t) =
    diffusion!(dv, v, cuboid, property, boundary)
```

Listing 4: Define interface for ODE solver for cooling-down process.

The cooling-down process is not depicted here because we focus on the heating-up process as described next.

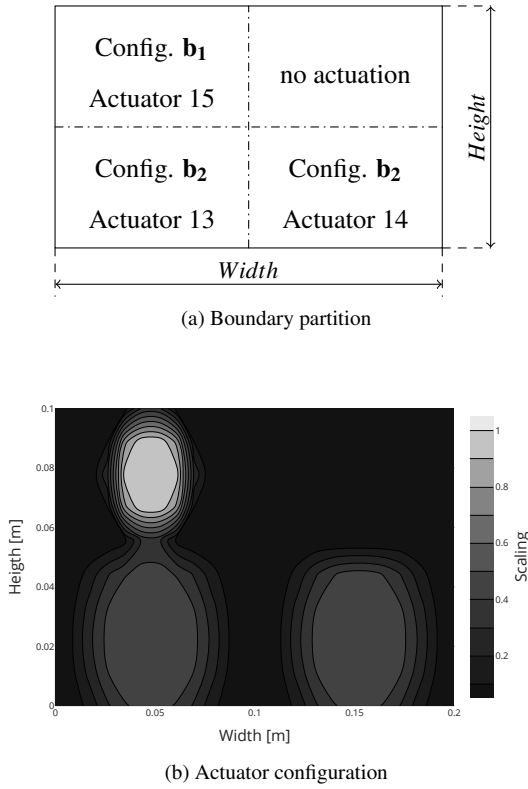


Figure 1: Partition (a) and configuration (b) on east boundary at $x_1 = L$ with configurations $b_1: (m_1, M_1, v_1) = (1.0, 50 \text{ I}_{3 \times 3}, 3)$ and $b_2: (m_2, M_2, v_2) = (0.5, 30 \text{ I}_{3 \times 3}, 2)$.

Heating-up process

The heating-up process extends the previous steps by specifying the position and spatial configuration of actuators on the boundary sides as discussed in Section 2. In this example, we assume actuation on boundaries UNDERSIDE and EAST. The underside is subdivided in 4×3 partitions and for each of it an individual actuator with configuration b_1 where $m_1 = 1$, $M_1 = 50 \text{ I}_{3 \times 3}$, $v_1 = 3$ is specified as in Listing 5.

```
config = setConfiguration(1.0, 3, 50)
setActuation!( actuation, cuboid, (4,3),
               config, :underside )
```

Listing 5: Specify actuation on underside.

Boundary EAST is subdivided manually with 2×2 partitions, as portrayed in Figure 1, where two fields are defined by configuration b_2 with $m_2 = 0.5$, $M_2 = 30 \text{ I}_{3 \times 3}$, $v_2 = 2$, one field is defined by b_1 as noted above and one field is not actuated. See also the complete listing [10].

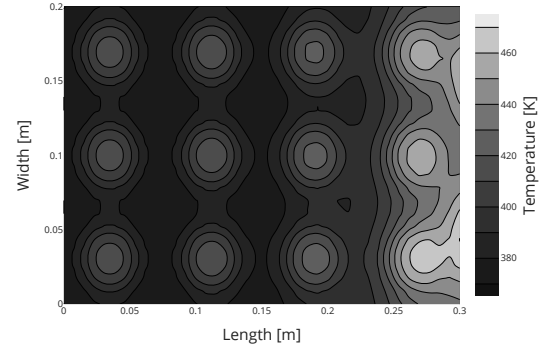


Figure 2: Temperature on the underside at $x_3 = 0$.

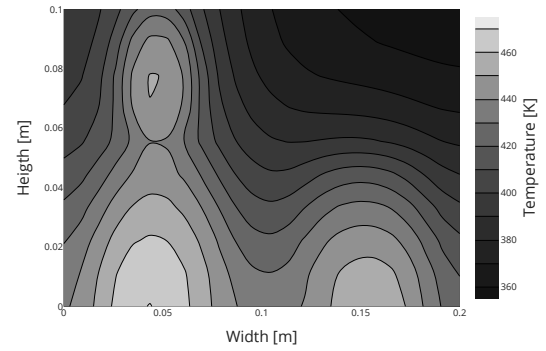


Figure 3: Temperature on the east boundary at $x_1 = L$.

A constant heat input $u_n(t) = 4 \cdot 10^5$ for actuator $n = 1, \dots, 15$ (12 actuators on the underside and 3 on the east boundary) is set and the ODE interface is implemented as in Listing 6.

```
u_in = 4e5 * ones(15)
heating_up!(dv, v, param, t) =
    diffusion!(dv, v, cuboid, property,
              boundary, actuation, u_in)
```

Listing 6: Define interface for ODE solver for heating-up process.

The heating-up process is simulated for $T_f = 200$ seconds and the final temperature distribution on the underside (at $x_3 = 0$) and east boundary (at $x_1 = L$) are portrayed in Figure 2 and 3. They unveil the strong influence of the actuator configuration on the resulting temperature distribution which reaches up to 470 Kelvin. A three-dimensional temperature distribution is illustrated in Figure 4 for temperatures above 360 Kelvin.

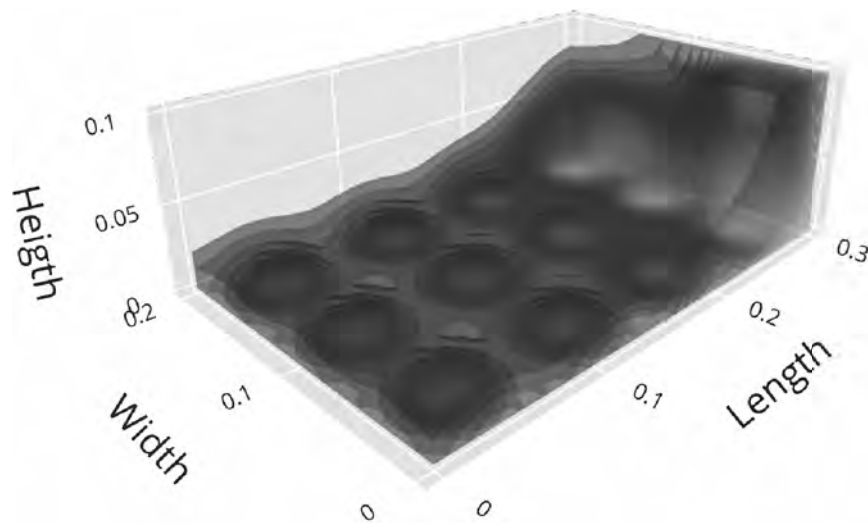


Figure 4: Temperature distribution in the cuboid for temperatures higher than 360 Kelvin.

4 Conclusion

We introduced the software library HESTIA.JL for modeling of three-dimensional heat conduction with boundary actuation. The recent version is able to approximate linear and quasi-linear (isotropic) heat conduction and to handle radial symmetric actuator configurations. Our further research will focus on the development of optimal control modules for HESTIA.JL, and on its good integration in Julia’s Scientific Machine Learning ecosystem.

References

- [1] Weller HG, Tabor G, Jasak H, Fureby C. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Computers in Physics*. 1998; 12(6): 620–631.
- [2] Logg A, Wells GN. DOLFIN: Automated finite element computing. *ACM Transactions on Mathematical Software*. 2010; 37(2), 1-28.
- [3] Schlottke-Lakemper M, Gassner GJ, Ranocha H, Winters AR, Chan J. Trixi.jl. Zenodo. 2022. Available: <https://zenodo.org/record/6372038>
- [4] Fuhrmann J, contributors. VoronoiFVM.jl: Finite volume solver for coupled nonlinear partial differential equations. Zenodo. 2022. Available: <https://doi.org/10.5281/zenodo.6151074>
- [5] Scholz S. Hestia.jl. GitHub. 2022. Available: <https://github.com/stephans3/Hestia.jl>
- [6] Bezanson J, Edelman A, Karpinski S, Shah VB. Julia: A fresh approach to numerical computing. *SIAM Review*. 2017; 59(1): 65-98.
- [7] Rackauckas C, contributors. SciML/DifferentialEquations.jl. Zenodo. 2022. Available: <https://doi.org/10.5281/zenodo.5837925>
- [8] Baehr HD, Stephan K. *Heat and mass transfer*. Springer Science & Business Media, 2011.
- [9] Scholz S, Berger L. Modeling of a multiple source heating plate. *arXiv preprint arXiv:2011.14939*. 2020.
- [10] Scholz S. HestiaDemonstration.jl. GitHub. 2022. Available: <https://github.com/stephans3/HestiaDemonstration.jl>

Untersuchung der anwendungsspezifischen Verifikations- und Validierungstechniken unter Berücksichtigung einer veränderten Datenbasis

Katharina Langenbach^{1*}, Anne Antonia Scheidler¹, Markus Rabe¹

¹ Fachgebiet IT in Produktion und Logistik, Technische Universität Dortmund, Leonhard-Euler-Str. 5, 44227 Dortmund, Deutschland; *katharina.langenbach@tu-dortmund.de

Abstract. Basierend auf der Weiterentwicklung von IT-Prozessen und der industriellen Globalisierung wächst der verfügbare Datenbestand kontinuierlich. Dieses Wachstum bedingt eine veränderte Datenbasis in den zugrundeliegenden IT-Systemen. Der vermehrte Einsatz von datenbasierten Anwendungen in der Industrie und Forschung verstärkt zudem die Notwendigkeit, Vertrauenswürdigkeit messbar zu gestalten. Ein elementarer Bestandteil der Messbarkeit sind Verifikations- und Validierungstechniken. Hierbei bietet das Anwendungsfeld ein erstes Unterscheidungskriterium für den Einsatz und die Anwendung der verwendeten Methoden. Etablierte Anwendungsfelder dieser Techniken sind Modellierung und Simulation, die Wissensentdeckung in Datenbanken und das Data Farming. Dieses Paper gibt einen Überblick über anwendungsspezifische Verifikations- und Validierungstechniken unter Berücksichtigung der Datenqualität und leitet daraus mögliche Forschungsfelder im Kontext einer veränderten Datenbasis ab.

Einleitung

Sowohl in vielen Bereichen der Industrie als auch in der Forschung werden immer mehr Daten erfasst, gespeichert und verarbeitet (Tanase et al., 2018; Steiner, 2017). Diese Daten werden für unterschiedliche Anwendungen und Analysen verwendet, auf deren Grundlage Entscheidungen abgeleitet werden. Um fundierte Entscheidungen treffen zu können, muss Vertrauen in datenbasierte (Analyse-) Anwendungen geschaffen werden. Verifikation und Validierung (V&V) unterstützt dabei, dieses Vertrauen zu etablieren.

Um die geänderte Datenbasis zu berücksichtigen, diskutiert der Beitrag auf Basis eines Überblicks über etablierte V&V-Techniken in unterschiedlichen Anwendungsfeldern möglichen Forschungsbedarf.

1 Grundlagen

Die V&V stammt ursprünglich aus der Softwareentwicklung, ist aber aufgrund des wachsenden Einsatzgebietes von Modellen und datenbasierten Anwendungen nicht auf diese beschränkt (Scheidler und Rabe, 2019; Hofmann, 2013). Allgemein erfolgt durch die V&V eine Kontrolle unterschiedlicher Qualitätsmerkmale von Daten, Modellen bzw. Anwendungen (Scheidler und Rabe, 2019; Rabe et al., 2008). Zur Durchführung der V&V müssen je nach angewandter Technik Datenqualitätsmerkmale wie z. B. Genauigkeit, Vollständigkeit, Korrektheit und Aktualität erfüllt sein (Olson, 2008; Wang und Strong, 1996). Je nach Anwendungsfeld sind die Qualitätsmerkmale von unterschiedlich großer Bedeutung, sodass auch die Wahl von V&V-Techniken, der Einsatz und die Bewertung von deren Ergebnissen entsprechend angepasst werden müssen.

Die Wahl einer V&V-Technik hängt nicht nur von ihrem Anwendungsfeld ab, sondern auch von der Art und Menge der zu betrachtenden Daten (Scheidler und Rabe, 2019; Rabe et al., 2008). Dabei hat sich der Umfang der zur Verfügung stehenden Input-Daten, beispielsweise im Zuge der Etablierung von Digital Twins oder Smart Factorys, in den letzten Jahren stark erhöht (Feng et al., 2020; Oussous et al., 2018; Tao et al., 2018). Die anfallenden Daten stammen zudem aus unterschiedlichen Datenquellen und sind vermehrt heterogen (Oussous et al., 2018). Diese Datenquellen stellen Daten überwiegend in hoher Geschwindigkeit zur Verfügung, was die Notwendigkeit einer schnellen Verarbeitung der Daten bedingt, sodass zeitnah anwendungsspezifische Reaktionen möglich sind. Dies stellt neue Anforderungen an die Datenhaltung, Verarbeitung und Analyse sowohl auf der Software- als auch auf der Hardwareseite (Oussous

et al., 2018; Z. Zheng et al., 2013). Daraus ergeben sich veränderte Anforderungen an die V&V, weil Daten zur Konzeptionierung von Anwendungen, für die V&V und für den Einsatz der Anwendungen benötigt werden (Sargent, 2008). Die zuvor aufgeführte veränderte Datenbasis bedingt eine Adaption der V&V-Anwendungsbereiche.

2 V&V in unterschiedlichen Anwendungsfeldern

Die V&V findet in unterschiedlichen Branchen Anwendung, beispielsweise in der Logistik, der Fertigung und dem Gesundheitswesen. Dabei werden in Abhängigkeit der Branche unterschiedliche Schwerpunkte in der V&V gesetzt. Weiter ergeben sich Unterschiede aus den Anwendungsfeldern der V&V. V&V wird nicht nur in der Softwareentwicklung angewandt, sondern etwa auch in der Modellierung und Simulation (M&S), dem Data Farming und der Wissensentdeckung in Datenbanken (KDD). Dabei ist die V&V häufig in den zugrundeliegenden Vorgehensmodellen verankert. Im Folgenden werden exemplarisch die drei oben genannten Anwendungsfelder der V&V betrachtet. M&S beschäftigt sich mit der vereinfachten Abbildung von Systemen auf Basis von Computerprogrammen (Galvao Scheidegger et al., 2018; Banks, 1998). Dabei dient die V&V der Prüfung, ob ein Modell korrekt und zudem für die Anwendung passend gebildet wurde (Roy und Oberkamp, 2011; Rabe et al., 2008). Das KDD generiert Wissen aus Input-Daten unter anderem durch Data Mining und der Interpretation von Mustern (Scheidler und Rabe, 2019; Fayyad et al., 1996). Die V&V stellt hierbei sicher, dass dem generierten Wissen vertraut werden kann. Das Data Farming verwendet Simulationsmodelle zur Datengenerierung, auf deren Basis Analysen durchgeführt werden können (Sanchez, 2019). Die V&V ist in diesem Fall verantwortlich für die Prüfung der Vertrauenswürdigkeit der durch das Data Farming erzeugten Daten.

Durch die kurze Beschreibung der drei Anwendungsfelder der V&V wird deutlich, dass sich die Zielsetzungen der V&V je nach Anwendungsfeld unterscheiden.

In **Tabelle 1** werden V&V-Techniken aus den unterschiedlichen Anwendungsfeldern in Kombination mit Anmerkungen bezüglich der zur sinnvollen Durchführung notwendigen Datenqualität aufgeführt. Die Techniken werden dabei zur besseren Übersichtlichkeit in Kategorien eingeteilt.

Die aufgeführten Handlungsfelder sollten nie getrennt voneinander betrachtet werden. So ist insbesondere das Feld M&S häufig Teil weiterer Anwendungsfelder wie des Data Farming und des KDD, da komplexe Systeme so formalisiert werden müssen, dass sie mit den vorhandenen Mitteln analysiert und verarbeitet werden können. Somit bauen die unterschiedlichen betrachteten Anwendungsfelder in Teilen aufeinander auf. Ferner ist zu berücksichtigen, dass nie eine einzelne V&V-Technik als ausreichend zu betrachten ist, da durch diese Techniken unterschiedliche Schwerpunkte gesetzt werden.

Ein weiterer Aspekt wird bei der Betrachtung der Anzahl von etablierten V&V-Techniken für die jeweiligen Anwendungsfelder deutlich. So existiert bei der M&S eine Vielzahl von Techniken, welche auf unterschiedlichen Anforderungen und Voraussetzungen ausgelegt sind. Im Data Farming sind hingegen kaum Techniken bekannt. Dort wird in wenigen Fällen die Validität der Ergebnisse durch den Vergleich mit Realwerten bzw. den Input-Daten überprüft. Dies erweckt den Eindruck, dass umso neuer ein Anwendungsfeld ist, desto weniger etablierte V&V-Techniken verfügbar sind.

Bei den V&V-Techniken in der M&S sind insbesondere die informellen Techniken auffällig, welche eine starke Abhängigkeit von der durchführenden Person aufweisen. Bei den eher als objektiv einzustufenden Techniken wird eine hohe Datenqualität vorausgesetzt, da die Techniken ansonsten kein aussagekräftiges Ergebnis liefern. Es ist herauszustellen, dass in allen Anwendungsfeldern die Datenqualität der Input-Daten eine Rolle spielt, da diese Daten die Grundlage für alle folgenden Schritte bilden. Dies legt nahe, dass auf diesen Bereich ein verstärktes Augenmerk gelegt werden sollte, insbesondere unter der Berücksichtigung einer sich ändernden Datenbasis.

3 Ausblick

Bei der Untersuchung der etablierten V&V-Techniken in verschiedenen Anwendungsfeldern hat sich gezeigt, dass sich die Menge unterschiedlicher V&V-Techniken je nach Anwendungsfeld unterscheidet. Ein mögliches Forschungsfeld ist die Bestimmung des notwendigen V&V-Umfangs in Abhängigkeit der zugrundeliegenden Glaubwürdigkeit der Anwendungsfelder und ihrer Elemente. Insbesondere ist zu prüfen, ob existierende V&V-Techniken Anwendung finden können, oder ob spezifische neue Techniken entwickelt werden müssen.

Tabelle 1: V&V-Techniken für verschiedene Anwendungsfelder unter Berücksichtigung der notwendigen zugrundeliegenden Datenqualitätskriterien (beispielhaft)

Anwendungsfeld	Kategorisierungen von V&V-Techniken	Datenqualitätskriterien	Beispielhafte Literatur
M&S	Informell (z. B. Audit, Schreibtischtest, Validierung im Dialog, Reviews)	Eignung Konsistenz Verständlichkeit	Rabe et al., 2008 Balci, 1997
	Statisch (z. B. Ursache-Wirkungs-Graph, Strukturelle Analyse, Syntax Analyse, Daten-Abhängigkeitsanalyse)	Konsistenz Korrektheit Plausibilität Vollständigkeit	Rabe et al., 2008 Balci, 1997
	Dynamisch (z. B. Black-Box-Test, Statistische Tests, Animation, Input Tests, Vergleichstest)	Eignung Korrektheit Vollständigkeit	Rabe et al., 2008 Balci, 1997
	Formell (z. B. Induktion, Lambda Calculus, Beweis der Korrektheit)	Eignung Korrektheit Vollständigkeit	Rabe et al., 2008 Balci, 1997
	Datenvalidität	Aktualität Eignung Korrektheit Vollständigkeit	Sargent, 2008
	Operationale Validität, Subjektiv (z. B. Graphischer Vergleich)	Eignung	Sargent, 2008
	Operationale Validität, Objektiv (z. B. Statistische Tests)	Eignung Genauigkeit Korrektheit Vollständigkeit	Sargent, 2008
Data Farming	Input-Daten-Spektrum	Vollständigkeit	Sanchez und Sánchez, 2017 Horne und Meyer, 2004
	Validierung von Mikroprozessen (z. B. durch Vergleiche mit Echtdaten)	Aktualität Eignung Plausibilität	Hofmann, 2013
	Validierung des zugrundeliegenden Simulationsmodells (s. M&S)	Siehe M&S	Schulze et al., 2018
KDD	Datenvalidität	Aktualität Eignung Korrektheit Vollständigkeit	Hunker et al., 2021
	Validierungsdatensatz (z. B. Statistische Techniken)	Eignung Machbarkeit Plausibilität	Scheidler und Rabe, 2019
	Dreiecksmodell	Unterschiedlich	Scheidler und Rabe, 2019

Weiter ist zu erkennen, dass die Ansprüche an die Datenqualität für unterschiedliche V&V-Techniken variieren. Insbesondere unter Berücksichtigung der veränderten Datenbasis ist dies von Interesse. Es ist zu untersuchen, inwieweit die Anwendbarkeit von V&V-

Techniken durch die veränderte Datenbasis beeinflusst wird. So scheinen beispielsweise manuelle im Gegensatz zu automatisierbaren Techniken für größere und heterogene Datenmengen weniger geeignet. Von einem anderen Standpunkt aus betrachtet, ergibt sich in diesem

Zusammenhang die Frage, wie sich die veränderte Datenbasis in bereits existierende Vorgehensmodelle, die V&V beinhalten, einbinden lässt und ob und in welcher Form dort Anpassungen notwendig sind.

Insgesamt wird deutlich, dass eine Betrachtung und Neubewertung der V&V und zugehörigen V&V-Techniken sowie bestehender Vorgehensmodelle unter Berücksichtigung einer veränderten Datenbasis sinnvoll sind, um die vertrauenswürdige und sinnvolle Nutzung datenbasierter Anwendungen zu gewährleisten.

Literatur

- [1] Balci, O. Verification, Validation and Accreditation of Simulation Models. In: Andradóttir, S.; Healy, K.J.; Withers, D.H.; Nelson, B.L. (Hrsg.): *Proceedings of the 1997 Winter Simulation Conference (WSC)*, Atlanta, Georgia, United States, 1997, S. 135–141.
- [2] Banks, J.: Principles of Simulation. In: Banks, J. (Hrsg.): *Handbook of Simulation*. New York, Chichester, Weinheim: Wiley 1998, S. 3–30.
- [3] Fayyad, U.M.; Piatetsky-Shapiro, G.; Smyth, P. Knowledge Discovery and Data Mining: Towards a Unifying Framework. *KDD* (1996) 96, S. 82–88.
- [4] Feng, J.; Li, F.; Xu, C.; Zhong, R.Y. Data-Driven Analysis for RFID-Enabled Smart Factory: A Case Study. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 50 (2020) 1, S. 81–88.
- [5] Galvao Scheidegger, A.P.; Banerjee, A.; Pereira, T.F. Uncertainty Quantification in Simulation Models: A Proposed Framework and Application through Case Study. In: Rabe, M.; Skoogh, A.; Mustafee, N.; Juan, A.A. (Hrsg.): *Simulation for a noble cause*, Gothenburg, Sweden, 12/9/2018 - 12/12/2018, 2018, S. 1599–1610.
- [6] Hofmann, M. Simulation-based Exploratory Data Generation and Analysis (Data Farming): A Critical Reflection on its Validity and Methodology. *The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology* 10 (2013) 4, S. 381–393.
- [7] Horne, G.E.; Meyer, T.E. Data Farming: Discovering Surprise. In: Ingalls, R.G. (Hrsg.): *Proceedings of the 2004 Winter Simulation Conference (WSC)*, Washington, D.C., 2004, S. 792–798.
- [8] Hunker, J.; Wuttke, A.; Scheidler, A.A.; Rabe, M. A Farming-for-Mining-Framework to Gain Knowledge in Supply Chains. In: Kim, S.; Feng, B.; Smith, K.; Masoud, S.; Zheng, Z. (Hrsg.): *Proceedings of the 2021 Winter Simulation Conference (WSC)*, Phoenix, AZ, USA, 2021, S. 1–12.
- [9] Olson, J.E. Data quality: The accuracy dimension. Amsterdam: Morgan Kaufmann 2008.
- [10] Oussous, A.; Benjelloun, F.-Z.; Ait Lahcen, A.; Belfkih, S. Big Data technologies: A survey. *Journal of King Saud University - Computer and Information Sciences* 30 (2018) 4, S. 431–448.
- [11] Rabe, M.; Spieckermann, S.; Wenzel, S. Verifikation und Validierung für die Simulation in Produktion und Logistik: Vorgehensmodelle und Techniken. Berlin, Heidelberg: Springer Berlin Heidelberg 2008.
- [12] Roy, C.J.; Oberkampf, W.L. A Comprehensive Framework for Verification, Validation, and Uncertainty Quantification in Scientific Computing. *Computer Methods in Applied Mechanics and Engineering* 200 (2011) 25-28, S. 2131–2144.
- [13] Sanchez, S.M. Data Farming: Better Data, Not Just Big Data. In: Mustafee, N.; Rabe, M.; Bae, K.-H.G.; Szabo, C.; Lazarova-Molnar, S. (Hrsg.): *Proceedings of the 2019 Winter Simulation Conference (WSC)*, National Harbor, MD, 2019, S. 425–439.
- [14] Sanchez, S.M.; Sánchez, P.J.: Better Big Data via Data Farming Experiments. In: *Advances in Modeling and Simulation*: Springer, Cham 2017, S. 159–179.
- [15] Sargent, R.G. Verification and Validation of Simulation Models. In: Mason, S.J. (Hrsg.): *Proceedings of the 2008 Winter Simulation Conference (WSC)*, Miami, FL, USA, 2008, S. 157–169.
- [16] Scheidler, A.A.; Rabe, M. Integral Verification and Validation for Knowledge Discovery Procedure Models. *International Journal of Business Intelligence and Data Mining* 1 (2019) 1, S. 1.
- [17] Schulze, T.; Feldkamp, N.; Bergmann, S.; Straßburger, S. Data Farming und simulationsbasierte Robustheitsanalyse für Fertigungssysteme. In: Decatcu, C.; Schramm, T.; Zobel, K. (Hrsg.): *Tagungsband ASIM - 24. Symposium Simulationstechnik*, Hamburg, 2018, S. 243–252.
- [18] Steiner, R. Grundkurs relationale Datenbanken: Einführung in die Praxis der Datenbankentwicklung für Ausbildung, Studium und IT-Beruf. Wiesbaden, Heidelberg: Springer Vieweg 2017.
- [19] Tanase, G.; Suzumura, T.; Lee, J.; Chen, C.-F.; Vijitbenjaronk, W.D.; Crawford, J.; Kanezashi, H.; Zhang, S. System G Distributed Graph Database 2018.
- [20] Tao, F.; Cheng, J.; Qi, Q.; Zhang, M.; Zhang, H.; Sui, F. Digital Twin-Driven Product Design, Manufacturing and Service with Big Data. *The International Journal of Advanced Manufacturing Technology* 94 (2018) 9-12, S. 3563–3576.
- [21] Wang, R.Y.; Strong, D.M. Beyond Accuracy: What Data Quality Means to Data Consumers. *Journal of Management Information Systems* 12 (1996) 4, S. 5–33.
- [22] Z. Zheng; J. Zhu; M. R. Lyu Service-Generated Big Data and Big Data-as-a-Service: An Overview. In: *2013 IEEE International Congress on Big Data*, 2013, S. 403–410.

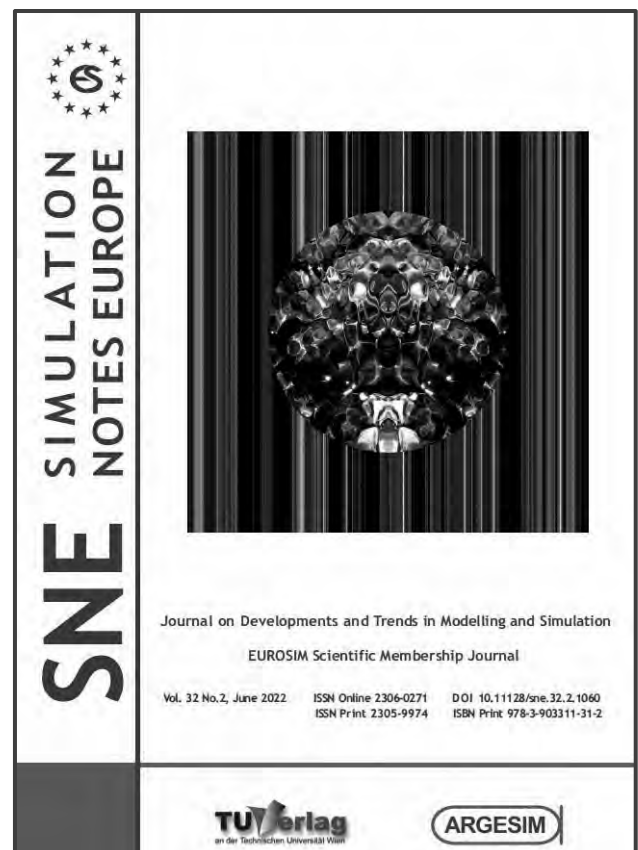
SNE SIMULATION NOTES EUROPE

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

SNE seeks to serve scientists, researchers, developers and users of the simulation process across a variety of theoretical and applied fields in pursuit of novel ideas in simulation and to enable the exchange of experience and knowledge through descriptions of specific applications. **SNE** puts special emphasis on the overall view in simulation, and on comparative investigations, as benchmarks and comparisons in methodology and application. Additionally, **SNE** welcomes also contributions in education in / for / with simulation.

SNE is the official membership journal of EUROSIM, the federation of European simulation societies and simulation groups, so **SNE** is open for postconference publication of contributions to conferences of the EUROSIM societies, and for special issues organized by EUROSIM societies.

SNE, primarily an electronic journal, follows an open access strategy, with free download in basic layout. Members of EUROSIM societies are entitled to download **SNE** in an elaborate and extended layout. Print **SNE** is available for specific groups of EUROSIM societies, and as print-on-demand from TU Verlag, TU Wien (www.tuverlag.at).

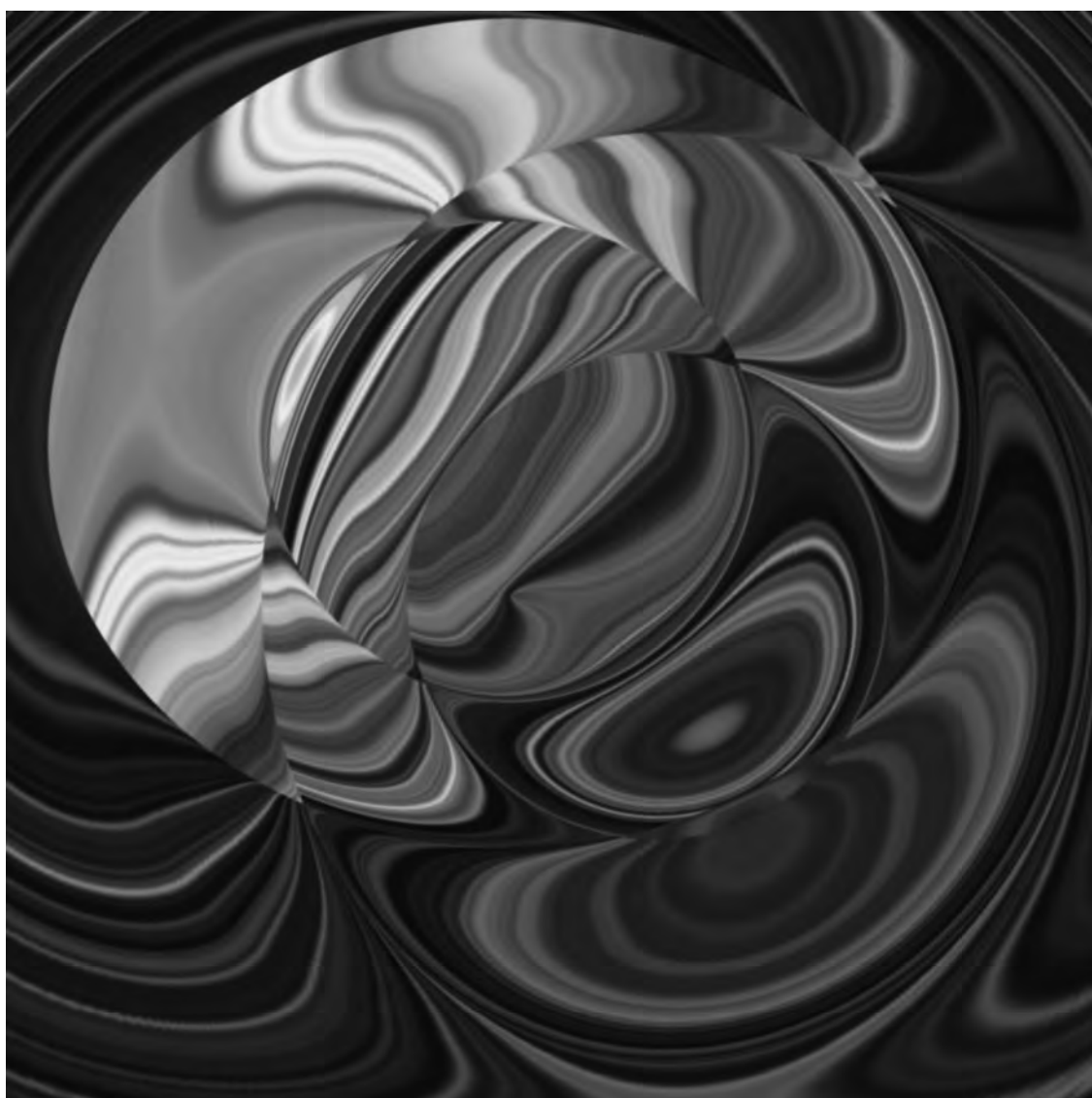


sne ournal org

ASIM SST 2022

TU Wien, 25.-27.7.2022

---- *eine Tagung wie es früher einmal war* ----



ALGORITHMIC ART BY VLATKO CERIC

VCERIC.NET

ISBN print
ISBN 978-3-901608-73-5
ARGESIM Verlag, Wien, 2022

ISBN ebook
ISBN 978-3-901608-96-4
ARGESIM Verlag, Wien, 2022

DOI ID
10.11128/arep.19