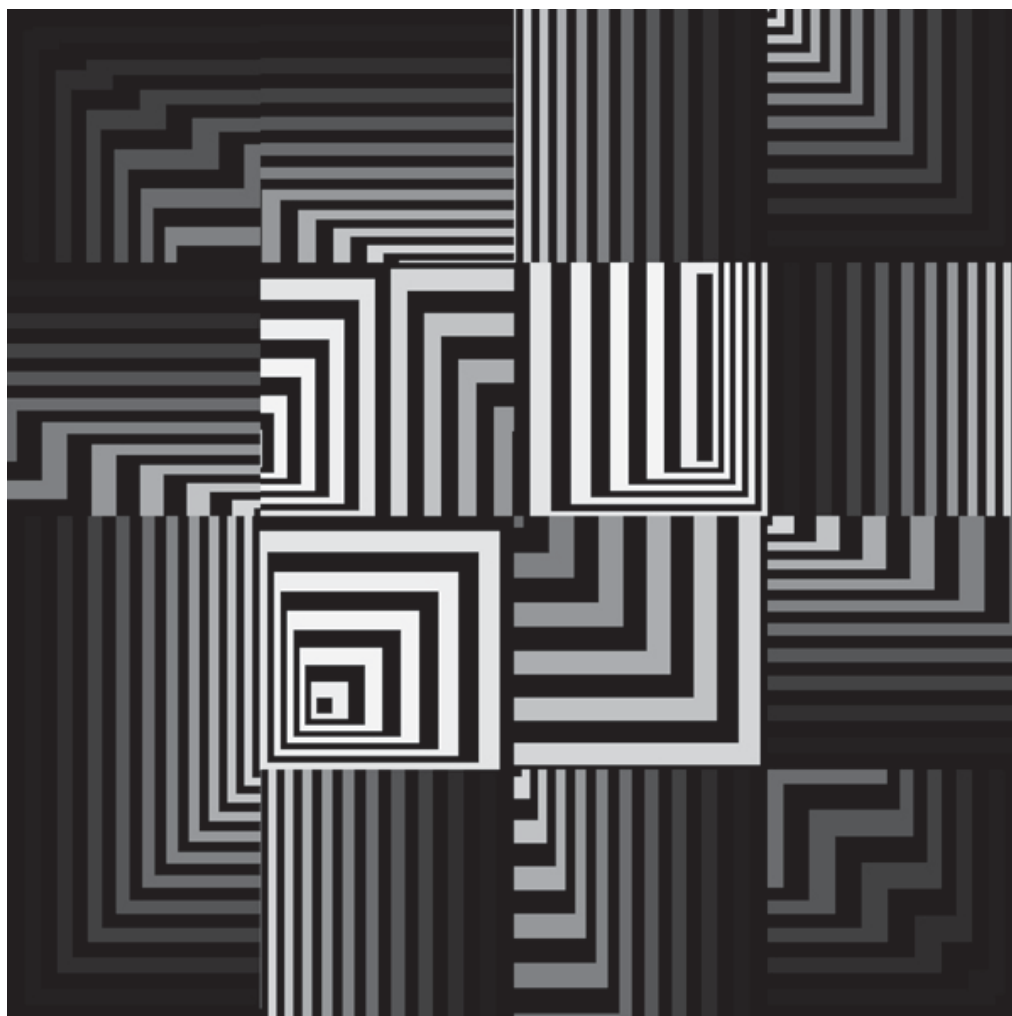




SNE SIMULATION NOTES EUROPE



Journal on Developments and Trends in Modelling and Simulation

EUROSIM Scientific Membership Journal

Vol. 30 No.3, Sept. 2020

ISSN Online 2306-0271

DOI 10.11128/sne.30.3.1052

ISSN Print 2305-9974

ISBN Print 978-3-903311-09-1

New Schedule for EUROSIM Conferences and Congress

Due to Corona Virus, also **EUROSIM** changed the schedule of **EUROSIM Conferences** and the **EUROSIM Congress** – all events will take place one year later. To bridge the 2020 conference gap the **EUROSIM** societies organise virtual conferences, and the **EUROSIM Board** starts in June 2020 **VESS** – the **Virtual EUROSIM Seminar**, a series of online presentations discussing trends in modelling and simulation and preparing emphasis of future **EUROSIM** events.


www.eurosim2023.eu

VESS - VIRTUAL EUROSIM SIMULATION SEMINAR
free online seminar series

The **EUROSIM Board** and **DBSS** start in June 2020 **VESS** – the **Virtual EUROSIM Seminar**, a series of online presentations discussing trends in modelling and simulation. These international online simulation seminars – monthly or bi-monthly – are open to everybody, via Zoom, lasting 60 minutes (45 minutes presentations, 15 minutes Q & A). Information and informal registration via website www.eurosim2023.eu



The First **SIMS EUROSIM Conference** on Modelling and Simulation, **SIMS EUROSIM 2021** takes place in Oulu, Finland, September 21-23, 2021. The 62nd International Conference of Scandinavian Simulation Society, SIMS 2021, is embedded with **SIMS EUROSIM 2021**. The **SIMS EUROSIM** conference will be organized every third year by **SIMS** and **EUROSIM**. The background of this conference series is in the 60-years history of Scandinavian Simulation Society, **SIMS**. The program of the **SIMS EUROSIM 2021 Conference** will have a multi-conference structure with several special topics related to methodologies and application areas. The program includes invited talks, parallel, special and poster sessions, exhibition and versatile technical and social tours – info www.scansims.org



MATHMOD organizers continue the conference series one year later, with **10th MATHMOD 2022**, February 16-18, 2022. **MATHMOD 2022**, one of **EUROSIM**'s main events, provides a forum for professionals, researchers, and experts in the field of theoretic and applied aspects of mathematical modelling for systems of dynamic nature. The scope of the **MATHMOD 2022** conference covers theoretic and applied aspects of various types of mathematical modelling (equations of various types, automata, Petri nets, bond graphs, qualitative and fuzzy models) for systems of dynamic nature (deterministic, stochastic, continuous, discrete or hybrid) – info and details www.mathmod.at



EUROSIM 2023, the **11th EUROSIM Congress**, will take place in Amsterdam, The Netherlands, Spring/Autumn 2023. It will be organized by the Dutch Benelux Simulation Society (www.dutchbss.org) supported mainly by their corporate members like TU Delft, Amsterdam University of Applied Sciences, EUROCONTROL and IGAMT (www.igamt.eu). Due to the growth of Simulation and its relationship with other analytical techniques like Big Data, AI, Machine Learning, Large Scale Simulation and others, the event will be structured, for the first time, in dedicated tracks focused on different areas and applications of Simulation ranging from aviation to health care and humanitarian activities. We have the ambition to attract at the congress participants from Academia, industry and governmental representatives to share the latest developments in Simulation and related activities and applications. Please follow the news and activities towards the **EUROSIM 2023** at www.eurosim2023.eu

Editorial

Dear Readers, SNE in times of Covid-19 virus – a new challenge for us. Due to the cancellation of conferences and workshops we had to re-schedule our publication plans, especially as foreseen post-conference publication are not available. We asked authors to submit announced publications earlier, we encouraged the review board to work with shorter deadlines, and so we are happy to finish also electronic SNE 30(3), September 2020, in time.

This issue continues the subject ‘high math meets simulation’ – Martin Holzinger presents his second contribution on PDE solution by conformal mappings dealing with conformal mappings from unit disk to general domains – to be continued and summarised by a surprising application in next year’s issue SNE 31(1).

The following two Technical Notes deal with modelling and simulation in production and logistics.

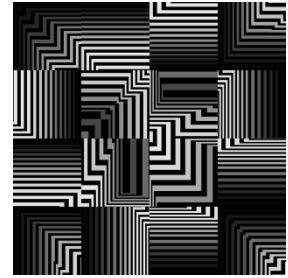
S. Wenzel et al. present logistic characteristic curves using discrete event simulation, and B. Heinzl and W. Kastner report on a case study with meta-heuristic simulation-based production planning for energy efficiency.

The fourth contribution of this issue by T. Pawletta et al. addresses advanced control: Q-Learning robot control for assembly cells using multiple agents, and SNE 30(3) concludes with an Educational Note by O. Ullrich and D. Lückerrath on modelling and simulation by cellular automata.

Thirty Years SNE – we are pleased that for SNE Volume 30 again Vlatko Čerčić, past president of the Croatian Simulation Society, provides his algorithmic art as design for SNE cover pages. The artist and simulationist Vlatko Čerčić has chosen four algorithmic art pictures from the series LABYRINTH for covers of SNE Volume 30. The cover of this issue SNE 30(3) presents LABYRINTH no. 17, and at right and preview of the algorithmic art picture for SNE 30(4) – LABYRINTH no. 21. For further info on Vlatko Čerčić’s graphics please visit www.vceric.net.

I would like to thank all authors for their contributions – and thanks to everybody for the extra work all had to do because of the Covid-19 circumstances, so that SNE 30(2) and SNE 30(3) could be published in time, and hopefully also the following SNE 30(4).

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



Contents SNE 30(3)

Online SNE 30(2), DOI 10.11128/sne.30.3.1052

ARGESIM Publisher, Vienna, www.argesim.org

Print SNE 30(3) ISBN 978-3-903311-09-1

TU Verlag Vienna, Print-on-Demand, www.tuverlag.at

High Precision Conformal Map of the Unit Disk to Star-Shaped Domains. <i>M. Holzinger</i>	85
Generating Logistic Characteristic Curves using Discrete Event Simulation and Response Surface Models. <i>S. Kuhnt, D. Kirchhoff, S. Wenzel, J. Stolipin</i>	95
Meta-heuristic Simulation-based Production Planning for Energy Efficiency: A Case Study. <i>B. Heinzl, W. Kastner</i>	105
Reduction of Complexity in Q-Learning a Robot Control for an Assembly Cell by using Multiple Agents. <i>G. Kunert, T. Pawletta, S. Hartmann</i>	117
Modeling and Simulation Using Cellular Automata. <i>O. Ullrich, D. Lückerrath</i>	125
EUROSIM Societies Short Info	N1 – N8
VESS – Virtual EUROSIM Simulation Seminar Series and EUROSIM Conferences	Cover Inside

SNE Contact & Info

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

→ www.sne-journal.org

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

✉ SNE Editorial Office

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

SNE SIMULATION NOTES EUROPE

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

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

Editor-in-Chief: Felix Breiteneker, TU Wien, Math. Modelling Group

✉ Felix.Breiteneker@tuwien.ac.at, ✉ eic@sne-journal.org

Print SNE and Print-on-Demand: TU-Verlag, Wiedner Hauptstrasse 8-10, 1040, Vienna, Austria – www.tuverlag.at

Publisher: ARGESIM ARBEITSGEMEINSCHAFT SIMULATION NEWS
c/o Math. Modelling and Simulation Group, TU Wien / 101,
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria;
www.argesim.org, ✉ info@argesim.org on behalf of
ASIM www.asim-gi.org and EUROSIM → www.eurosims.info
© ARGESIM / EUROSIM / ASIM 2020

SNE - Aims and Scope

Simulation Notes Europe (SNE) provides an international, high-quality forum for presentation of new ideas and approaches in simulation - from modelling to experiment analysis, from implementation to verification, from validation to identification, from numerics to visualisation - 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 follows the recent developments and trends of modelling and simulation in new and/or joining application areas, as complex systems and big data. SNE puts special emphasis on the overall view in simulation, and on comparative investigations, as benchmarks and comparisons in methodology and application. For this purpose, SNE documents the ARGESIM Benchmarks on *Modelling Approaches and Simulation Implementations* with publication of definitions, solutions and discussions. SNE welcomes also contributions in education in/for/with simulation.

A News Section in SNE provides information for EUROSIM Simulation Societies and Simulation Groups.

SNE, primarily an electronic journal, follows an open access strategy, with free download in basic layout. SNE is the official membership journal of EUROSIM, the *Federation of European Simulation Societies and Simulation Groups* – www.eurosim.info. Members of EUROSIM societies are entitled to download SNE in an elaborate and extended layout, and to access additional sources of benchmark publications, model sources, etc. Print SNE is available for specific groups of EUROSIM societies, and starting with Volume 27 (2017) as print-on-demand from TU Verlag, TU Wien. SNE is DOI indexed by CrossRef, identified by DOI prefix 10.11128, assigned to the SNE publisher ARGESIM (www.argesim.org).

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

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

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

- TN Technical Note, 6 – 10 p.
- PN Project Note 6 – 8 p.
- SW Software Note, 4 – 6 p.
- ON Overview Note – only upon invitation, up to 14 p.
- EN Education Note – 6 – 8 p.
- SN Short Note, max. 6 p.
- BN Benchmark Note, 2 – 8 p.
- EBN Edu B.Note, 2 10 p
- STN Student Note, 4-8 p, inv.

Further info and templates (doc, tex) at SNE's website.

www.sne-journal.org

SNE Editorial Board

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

Felix Breitenecker, Felix.Breitenecker@tuwien.ac.at
TU Wien, Math. Modelling, Austria, Editor-in-chief
David Al-Dabass, david.al-dabass@ntu.ac.uk,
Nottingham Trent University, UK
Maja Atanasijevic-Kunc, maja.atanasijevic@fe.uni-lj.si
Univ. of Ljubljana, Lab. Modelling & Control, Slovenia
Aleš Belič, ales.belic@sandoz.com
Sandoz / National Inst. f. Chemistry, Slovenia
Peter Breedveld, P.C.Breedveld@el.utwente.nl
University of Twente, Netherlands
Agostino Bruzzone, agostino@itim.unige.it
Università degli Studi di Genova, Italy
Francois Cellier, fcellier@inf.ethz.ch, ETH Zurich, Switzerland
Vlatko Čerić, vceric@efzg.hr, Univ. Zagreb, Croatia
Russell Cheng, rhc@maths.soton.ac.uk
University of Southampton, UK
Roberto Cianci, cianci@dime.unige.it,
Math. Eng. and Simulation, Univ. Genova, Italy
Eric Dahlquist, erik.dahlquist@mdh.se, Mälardalen Univ., Sweden
Umut Durak, umut.durak@dlr.de
German Aerospace Center (DLR) Braunschweig, Germany
Horst Ecker, Horst.Ecker@tuwien.ac.at
TU Wien, Inst. f. Mechanics, Austria
Vadim Engelson, vadime@mathcore.com
MathCore Engineering, Linköping, Sweden
Peter Groumpos, groumpos@ece.upatras.gr
Univ. of Patras, Greece
Edmond Hajrizi, ehajrizi@ubt-uni.net
University for Business and Technology, Pristina, Kosovo
Glenn Jenkins, GLJenkins@cardiffmet.ac.uk
Cardiff Metropolitan Univ., UK
Emilio Jiménez, emilio.jimenez@unirioja.es
University of La Rioja, Spain
Esko Juuso, esko.juuso@oulu.fi
Univ. Oulu, Dept. Process/Environmental Eng., Finland
Kaj Juslin, kaj.juslin@enbuscon.com, Enbuscon Ltd, Finland
Andreas Körner, andreas.koerner@tuwien.ac.at
TU Wien, Math. E-Learning Dept., Vienna, Austria
Francesco Longo, f.longo@unical.it
Univ. of Calabria, Mechanical Department, Italy
Yuri Merkuryev, merkur@itl.rtu.lv, Riga Technical Univ.
David Murray-Smith, d.murray-smith@elec.gla.ac.uk
University of Glasgow, Fac. Electrical Engineering, UK
Gasper Music, gasper.music@fe.uni-lj.si
Univ. of Ljubljana, Fac. Electrical Engineering, Slovenia
Thorsten Pawletta, thorsten.pawletta@hs-wismar.de
Univ. Wismar, Dept. Comp. Engineering, Wismar, Germany
Niki Popper, niki.popper@dwh.at, dwh Simulation Services, Austria
Kozeta Sevrani, kozeta.sevrani@unitir.edu.al
Univ. Tirana, Inst. f. Statistics, Albania
Thomas Schriber, schriber@umich.edu
University of Michigan, Business School, USA
Yuri Senichenkov, sneyb@dcn.infos.ru
St. Petersburg Technical University, Russia
Michal Štepanovský, stepami9@fit.cvut.cz
Technical Univ. Prague, Czech Republic
Oliver Ullrich, oliver.ullrich@iais.fraunhofer.de
Fraunhofer IAIS, Germany
Siegfried Wassertheurer, Siegfried.Wassertheurer@ait.ac.at
AIT Austrian Inst. of Technology, Vienna, Austria
Sigrid Wenzel, S.Wenzel@uni-kassel.de
Univ. Kassel, Inst. f. Production Technique, Germany
Grégory Zacharewicz, gregory.zacharewicz@mines-ales.fr
IMT École des Mines d'Alès, France

High Precision Conformal Map of the Unit Disk to Star-Shaped Domains

Martin Holzinger

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

SNE 30(3), 2020, 85-94, DOI: 10.11128/sne.30.tn.10521
Received: July 15, 2020; Revised: August 20, 2020;
Accepted: August 25, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Having the conformal map from unit square to unit disk at hand, we ask ourselves for a way to numerically map the disk to more general domains. Once such domains are parametrized by the square, knowledge of metric quantities flows in by the derivatives of the map and simulation of PDEs on such domains can easily be achieved. One way to numerically construct the map for star-shaped regions yields over solving Theodorsen's integral equation which establishes the boundary correspondence of angles. Focusing on highly accurate solutions, we present *Mathematica* test implementations and results for maps from unit disk to inverted ellipse (a), unit square (b) and onto a more general domain (c). Finally, the metric impact of the conformal map on the PDE itself is being investigated to enlighten the process of correcting spacial Finite-Difference approximations in general.

Introduction

In [4] the CTDS-Method has been used to solve a heat conduction problem on the unit square. Moreover, we showed that using elliptic functions to conformally map the square to the unit disk provides a way to treat partial differential equations aside regular structured grids necessary for CTDS. In an alternative point of view, by conformal parametrization of the disk by the square this regularity is being kept up - the map thus is not interpreted as point transformator but rather as an introducer of curvi-linear co-ordinates. Following this path, we investigate ways to numerically construct conformal maps from unit disk to more general domains.

1 THEODORSEN'S Equation

Suppose you are interested in finding the conformal map $\zeta = \xi + i\eta = f(w) = f(u + iv)$ from unit disk \mathfrak{E} onto a domain \mathfrak{G} bounded by a closed, piecewise Jordan arc \mathfrak{C} with polar representation $\rho(\theta)$. For uniqueness we demand $f(0) = 0$, $f'(0) > 0$ and moreover, let \mathfrak{G} be *star-shaped* with respect to $\zeta_0 = 0$,

$$\zeta_0 + t(\hat{\zeta} - \zeta_0) \in \mathfrak{G}, \quad \forall t \in [0, 1), \quad \forall \hat{\zeta} \in \mathfrak{C}.$$

GAIER [3] presents an elegant way of construction by first deriving an equation for the (real) *function of boundary correspondence* which links the angle $\varphi \in [0, 2\pi)$ on \mathfrak{E} to the angle $\theta(\varphi)$ describing \mathfrak{C} . For f regular on \mathfrak{E} and continuous on the closure $\overline{\mathfrak{E}}$ he concludes

$$\eta(e^{i\varphi}) = \eta(0) + \frac{1}{2\pi} \int_{0(p.v.)}^{2\pi} \xi(e^{i\vartheta}) \cot \frac{\varphi - \vartheta}{2} d\vartheta.$$

Introducing the help function $F(w) := \ln \frac{f(w)}{w}$,

$$F(w) = \ln |f(w)| - \ln |w| + i \arg(f(w)) - i \arg(w)$$

can be derived and putting all together results in

$$\theta(\varphi) = \varphi + \frac{1}{2\pi} \int_{0(p.v.)}^{2\pi} \ln \rho(\theta(\vartheta)) \cot \frac{\varphi - \vartheta}{2} d\vartheta$$

known as THEODORSEN's nonlinear and singular integral equation for establishing the boundary correspondence. For existence and uniqueness of a solution, $\rho(\theta)$ in $[0, 2\pi]$ absolute continuous and $|\sigma(\theta)| := |\rho'(\theta)/\rho(\theta)| \leq \varepsilon < 1$ almost everywhere are sufficient conditions. Latter is the so-called ε -condition.

As will be considered later, even in case of $\varepsilon > 1$ numerical algorithms are able to produce feasible results if carefully picked. On the other hand, the shape of the concrete geometry has heavy impacts on the quality of the numerical approximation, e.g. if \mathfrak{C} contains corners.

2 Approximation Methods for THEODORSEN'S Equation

In order to achieve a high-precision approximation for $\theta(\varphi)$, the function of boundary correspondence, for continuous and 2π -periodic real functions $h(\varphi)$ let us first introduce the operator

$$\mathcal{K}[h](\varphi) = h^*(\varphi) := \frac{1}{2\pi} \int_{0(p,v)}^{2\pi} h(\vartheta) \cot \frac{\varphi - \vartheta}{2} d\vartheta.$$

Here GAIER summarizes in [3] that the linear operator $\mathcal{K} : h \mapsto h^*$ has some properties *très chic*, namely producing an again 2π -periodic, continuous function $h^*(\varphi)$, whose Fourier series can be gained immediately out of the series for $h(\varphi)$: If we have an expansion

$$h(\varphi) = \frac{a_0}{2} + \sum_{n=1}^{\infty} (a_n \cos n\varphi + b_n \sin n\varphi)$$

at hand, we can deduce the series

$$h^*(\varphi) = \sum_{n=1}^{\infty} (a_n \sin n\varphi - b_n \cos n\varphi)$$

for the so-called *conjugate function* $h^*(\varphi)$. Having now a closer look at THEODORSEN'S equation, with $h = \ln \rho(\theta)$ it re-writes as $\theta(\varphi) = \varphi + \mathcal{K}[\ln \rho(\theta)](\varphi)$ and a Fourier sum for the nonlinear and singular integral can be applied. In fact, Fourier synthesis of $\ln \rho(\theta)(\varphi)$ turns out to be a very powerful method for our purpose of numerically constructing conformal maps.

Fix point iteration. One way to numerically treat THEODORSEN'S equation is by means of successive approximation. The 'fix point' in question is in our case a monotonically increasing function as a solution of the functional fix point problem

$$\theta = \iota + \mathcal{K}[\ln \rho(\theta)] = \mathcal{F}(\theta).$$

Formally, the iteration process is then defined by

$$\theta_{n+1} = \mathcal{F}(\theta_n), \quad n = 0, 1, 2, \dots,$$

and the iteration itself can by default be started with $\theta_0 = \iota$, the identity function (which trivially solves the problem for $\mathfrak{C} = \mathfrak{E}$). Of course any other more apt θ_0 can be used instead if at hand, e.g. any established boundary correspondence function $\theta(\varphi)$ for a similar shaped geometry \mathfrak{C} would be predestined for such an intention.

Newton's method. The well-known quadratic behavior when iterating the roots of real functions gives rise to reformulate the method to a functional zeroing problem. With $\psi(\varphi) = \theta(\varphi) - \varphi$, THEODORSEN'S equation reads $\psi = \mathcal{K}[\ln \rho(\psi + \iota)]$, hence [7, p.235]

$$\mathcal{F}(\psi) = \psi - \mathcal{K}[\ln \rho(\psi + \iota)] \equiv 0$$

has to be solved which can be formally iterated via

$$\psi_{n+1} = \psi_n - [\mathcal{F}'(\psi_n)]^{-1} \mathcal{F}(\psi_n), \quad n = 0, 1, 2, \dots,$$

whereas the statement concerning the starting function, $\psi_0 \equiv 0$ **or better if available**, turns in fact out to be essential for convergence of the iteration process.

To get the iteration method ready for implementation, one still has to resolve the role played by the operator \mathcal{F} . According to HÜBNER [6], this can be done by investigating its FRÉCHET derivative \mathcal{F}' while its necessary inversion $[\mathcal{F}']^{-1}$ is handled by solving a RIEMANN-HILBERT problem. Using these results, WEGMANN presents in [9] a formidable discrete-type algorithm which we made use of in our *Mathematica* implementations: Starting with an even (let $N = 2n$) decomposition of $[0, 2\pi)$, hence $\varphi^{(k)} = \frac{k\pi}{n}$, $k = 0, \dots, N-1$, and a given vector $\theta_0 = \varphi$ or better, \mathcal{K} is initially replaced by its discrete equivalent, the WITTICH operator

$$\mathcal{K}_N[h] = \sum_{j=1}^{n-1} a_j \sin j\varphi - b_j \cos j\varphi, \quad \text{where}$$

$$a_j = \frac{1}{n} \sum_{k=1}^N h^{(k)} \cos j\varphi^{(k)}, \quad b_j = \frac{1}{n} \sum_{k=1}^N h^{(k)} \sin j\varphi^{(k)}.$$

Note that all appearing vectors are set in bold type. With these preliminary remarks a WEGMANN step reads

$$\theta_{i+1} = \theta_i - \frac{f}{r^2} - \frac{(p + \hat{q} \cdot \tan \hat{v}) \cdot \exp(w)}{r},$$

the remaining task is calculating the auxiliary quantities (component-by-component, code sample see page 94):

$$f = \theta_i - \varphi - \mathcal{K}_N[\ln \rho(\theta_i)], \quad r = \sqrt{1 + \sigma^2(\theta_i)},$$

$$v = \arctan \sigma(\theta_i), \quad w = \mathcal{K}_N[v], \quad \hat{v} = \frac{1}{N} \sum_{k=1}^N v^{(k)},$$

$$q = \frac{f \cdot \sigma(\theta_i)}{r \cdot \exp(w)}, \quad p = \mathcal{K}_N[q], \quad \hat{q} = \frac{1}{N} \sum_{k=1}^N q^{(k)}.$$

3 Continuation on Unit Disk

As soon as $\theta(\varphi)$, the function of boundary correspondence is numerically determined, the remaining task consists in finding a way to get hold of $f(w)$, the conformal map from \mathfrak{E} to \mathfrak{G} . However the numeric construction of $\theta(\varphi)$ was achieved in detail, let us here merely suppose that this function is provided by the implementation as a continuous function.

In case that we made use of conjugate functions within our iteration process, the discrete Fourier coefficients a_k and b_k of $\ln \rho[\theta(\varphi)]$ can even be used for a continuation to the interior of the unit disk: Solve a DIRICHLET boundary value problem [5, p.27f.] to get

$$\zeta = f(w) = w \cdot \exp \left[\frac{a_0}{2} + \sum_{k=1}^{\infty} (a_k - ib_k) w^k \right].$$

This completes the elegant method GAIER showed us some sixty years ago. We note that Fourier synthesis of $\ln \rho[\theta(\varphi)]$ is the nucleus of the algorithm.

It is next to be stated that the underlying discrete trigonometric interpolation can only be expected to be acceptably accurate when the shape of the geometry to be mapped endorses this approach. In particular will an occurrence of vertices result in discontinuities of $\sigma(\theta)$ being fatal for this algorithm.

On the other hand, on smooth geometries even in case of $\varepsilon > 1$ WEGMANN'S algorithm can be used to produce highly accurate solutions to a certain degree, with convergence orders then falling back from quadratic to linear which is in fact a matter of computational time.

Aside trigonometric interpolation, SCHWARZ or CAUCHY formulae can be used to snatch the map, with $\theta(\varphi)$ known the former results in

$$f(w) = w \cdot \exp \left[\frac{1}{2\pi} \int_0^{2\pi} \ln \rho(\theta(\vartheta)) \frac{e^{i\vartheta} + w}{e^{i\vartheta} - w} d\vartheta \right]$$

whereas in the latter case f can be expressed as

$$f(w) = w \cdot \exp \left[\frac{1}{2\pi} \int_0^{2\pi} \frac{\ln \rho(\theta(\vartheta)) + i(\theta(\vartheta) - \vartheta)}{1 - we^{-i\vartheta}} d\vartheta \right].$$

Mathematica handles the evaluation of those complex-valued integrals with ease. Finally, from theory it is known that the overall error maxima of the numerical approximation will occur at the boundary. With this preparations, we are ready to do some numerical experiments and check the validity of our hitherto remarks.

4 Numerical Experiments

Let us first put our focus on two geometries with well-known closed-form solutions for the function of boundary correspondence as well as for the conformal map itself and then study the methods for a more general geometry where symmetry should hold with respect to the η -axis. Domain boundary points can be prescribed and its radius function can then be deduced by interpolation.

Aside the conformal map also its derivatives play a major part in a bigger context of our considerations and we shall therefore include at least numerical error examinations of first derivatives in our investigations.

4.1 Inverted ellipse

With $\rho(\theta) = \sqrt{1 - (1 - p^2) \cos^2 \theta}$ describing the boundary ($0 < p < 1$) the solution is available in closed-form, we have [3, p.264]

$$\theta(\varphi) = \arctan(p \tan \varphi)$$

for the function of boundary correspondence and

$$\zeta = f(w) = \frac{2pw}{1 + p + (1 - p)w^2}$$

for the conformal map itself. To set out to coding, we implement a basic fix point iteration code shown on the next page. Our intention is to keep things simple here to avoid programming errors and move on to more complex implementations when results are satisfying. As you can see, $\theta(\varphi)$ is implemented as a continuous function, trigonometric interpolation being carried out after evaluating $\ln \rho$ at 256 points using `LinearModelFit` throughout for the sake of higher working precision.

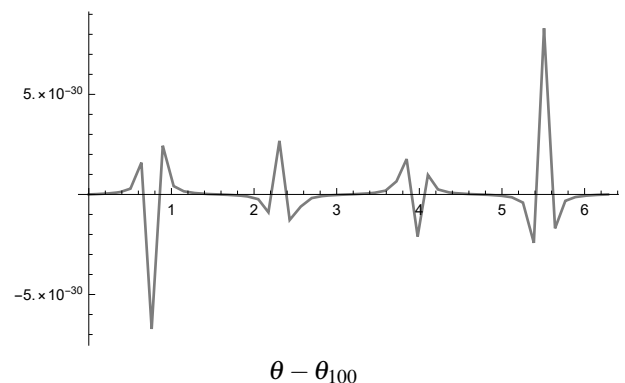


Figure 1: Verifying the base implementation: Calculating 100 fixpoint loops with $2n = 256$ discretization points.

```
(* Inverted Ellipsis: Crude Fixpoint Iteration Code as a starting point *)
p=6/10;zn=256;n=zn/2;r[phi_]=Sqrt[1-(1-p^2)Cos[phi]^2];lnR[phi_]=Log[r[phi]];
niter=100;basis=Table[Cos[2m*phi],m,0,zn/4];thetaneu[phi_]=phi;
Do[
  lnRv=Table[phi,lnR[thetaneu[phi]],phi,0,2Pi-Pi/n,Pi/n];
  lmfitlnR=LinearModelFit[lnRv,basis,phi,WorkingPrecision->80];
  lnRapprox[phi_]=lmfitlnR["BestFit", WorkingPrecision->80];
  anbeta=Coefficient[lnRapprox[phi],Table[Cos[m*phi],{m,1,n}]];
  klnRtrigapprox[phi_]=Sum[anbeta[[j]]*Sin[j*phi],{j,1,n}];
  thetaneu[phi_]=phi+klnRtrigapprox[phi], {i,1,niter} ]
```

Of course we can do much better with WEGMANN'S algorithm which is predestined for smooth geometries like the inverted ellipse, with $p = 0.6$ an ε -condition $\max|\sigma(\theta)| = 8/15 < 1$ results. Using η -symmetry, discrete Fourier transform [5, p.40] and external calculation of multiple used structures, or code on page 94 runs fast as lightning straight down to predefined `WorkingPrecision` within seven iteration steps (15s in total using $2n = 1024$ points), cf. Figure 2.

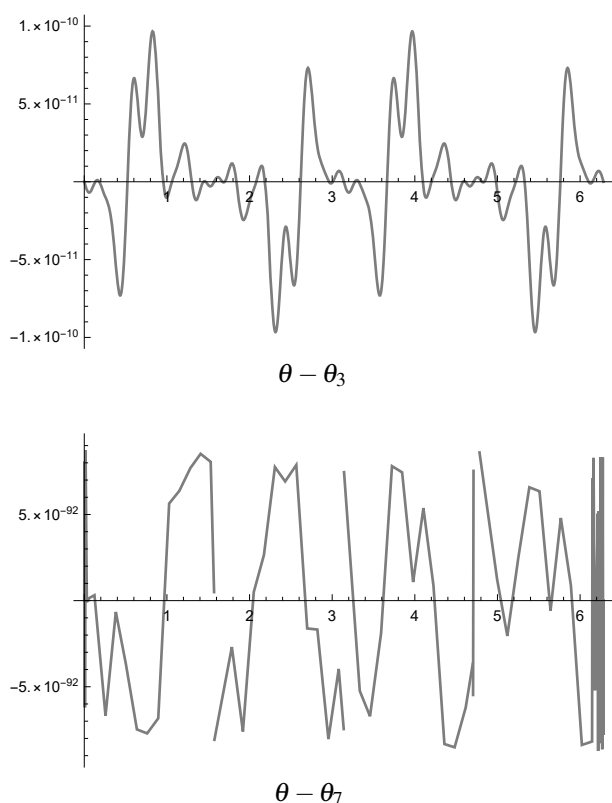


Figure 2: Approximation errors using WEGMANN'S method.

Figure 3 shows the results obtained when joining the map calculated with the trigonometric series for the unit disk and the map used in [4] transforming square to disk with elliptic functions. Note that computational errors for the derivatives show a very similar behavior.

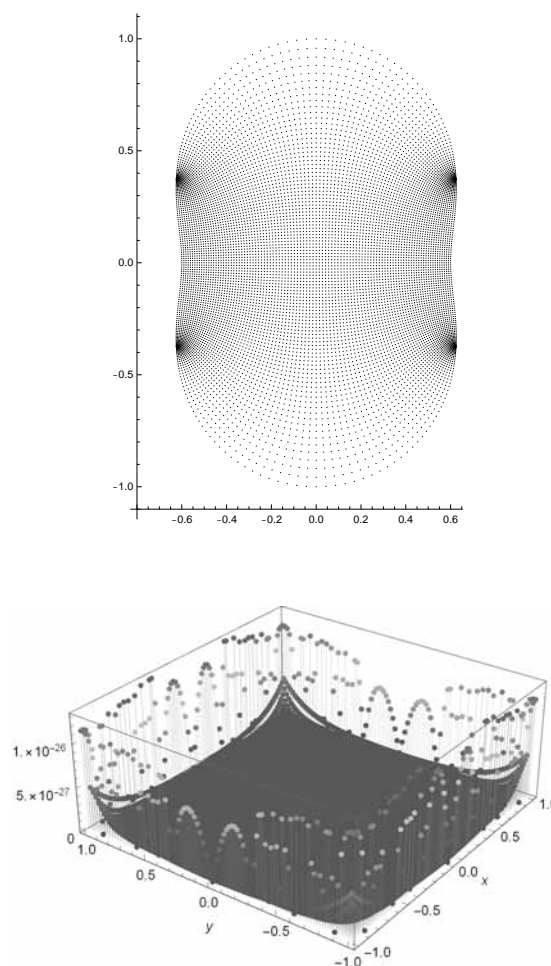


Figure 3: Mapping square to ellipse (t). Error behavior (b).

4.2 Unit square

Mapping the disk back to the square numerically is of particular interest. While all involved functions are known in closed form [3, p.265] and the square can be described via $\rho(\theta) = \sec(\theta)$, $\theta \in [-\frac{\pi}{4}, \frac{\pi}{4}]$, trigonometric approximation will be bound to fail with respect to numerical accuracy when constructing the function of boundary correspondence. We shall use the closed-form functions only for comparison purposes here.

Main difficulties arise primarily caused by the presence of corner singularities. For a piecewise smooth closed Jordan arc BLÜMLINGER [1, p.22] showed that in a vertex point $\zeta_i = f(w_i)$ with interior turning angle α_i/π , one can extract the singularity thus the derivative has a representation $\theta'(\varphi) = |\varphi - \varphi_i|^{\alpha_i-1} \lambda(\varphi)$. Using symmetries, on $[-\frac{\pi}{4}, \frac{\pi}{4}]$ we can try an even approach

$$\theta'(\varphi) = \left(\frac{\pi^2}{16} - \varphi^2\right)^{-\frac{1}{2}} \sum_{n=0}^{\infty} a_{2n} \varphi^{2n}.$$

A similar discussion can be found in [7, p.327, Theorem (12.2.1)]. Digressing trigonometric functions and conjugate expansions, RICHARDSON presents a coding sample in [8, p.471ff.] which we were able to refine using base functions $(\pi^2/16 - \varphi^2) \arcsin^{2n-1}(4\varphi/\pi)$ and $(\pi^2/16 - \varphi^2)^{(2n-1)/2} \arcsin(4\varphi/\pi)$ for an approximation of the function of boundary correspondence itself. In addition, only partial integration has been applied.

Figure 4 shows the resulting error, after 60 iteration steps with RICHARDSON's method the algorithm stagnates. Taking his approximation for θ (with error 10^{-7}) as a starting function, feeding it in our procedure and doing another 40 iteration steps, an error 10^{-11} results.

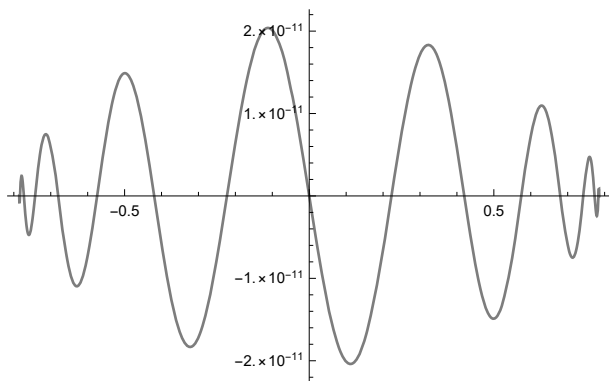


Figure 4: Approximation error $\theta - \theta_{100}$ on $[-\frac{\pi}{4}, \frac{\pi}{4}]$.

We see that, in contrary to the smooth situation of the inverted ellipse in the section before, neither Newton's method nor the usage of trigonometric conjugation is an option to snatch the operator $\mathcal{K}[\ln \rho(\theta)]$ in this case.

Moreover, the idea behind the algorithm is to attack the integral in Theodorsen's equation directly within the fix point loop using `NIntegrate` and a handful of discretization points. Tell *Mathematica* the angular location of the singularities within the integration routine and you can expect no further complaints. As before, `LinearModelFit` is used to gain a continuous representation of θ , interpolation now being done by replacing sin-/cos-functions with the new base functions.

Figure 5 shows that we can even control the derivative error in the vicinity of the singularity while Figure 6 illustrates the overall error when numerically parametrizing the unit square by itself. The interior of the disk was in this case treated with CAUCHY formula.

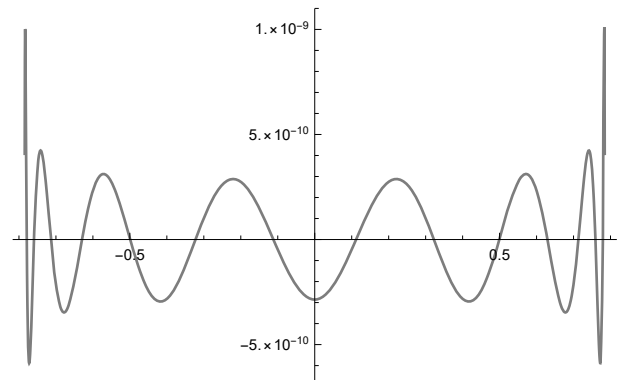


Figure 5: Approximation error $\theta' - \theta'_{100}$ on $[-\frac{\pi}{4} + 10^{-4}, \frac{\pi}{4} - 10^{-4}]$.

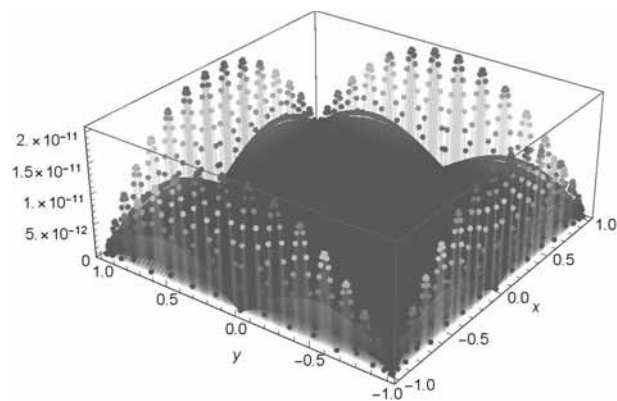


Figure 6: A previously exact mapped 201×201 -grid on the square is numerically mapped back from the unit disk. Expect the error maxima on the boundary.

4.3 General axially symmetric domains

For more general star-shaped domains we first implement a continuous boundary description function $R(\theta)$ by means of trigonometric¹ or Hermite² interpolation. In *Mathematica* this can easily be achieved by creating a list `geo` containing the prescribing boundary points (cf. Table 1 with η -symmetric shape definitions for three sample geometries). See Figure 8 for the resulting geometries and their corresponding ε -conditions. Also note that we are thus able to provide a radius value to any desired angular argument θ .

With such a provisioning of the radius function $R(\theta)$ we can now make use of one of the above mentioned iteration processes: Fix point approximation or NEWTON iteration in either its continuous or discrete form to get hold of the function of boundary correspondence.

As we are lacking an exact solution $\theta(\varphi)$ for comparison in more general cases, the iteration differences $E_i(\varphi) := \theta_i(\varphi) - \theta_{i-1}(\varphi)$ can be observed in case that the iterations are carried out with continuous functions $\theta_i(\varphi)$ and, likewise for discrete iterations, a discrete error vector can be defined via $e_i^{(k)} := \theta_i^{(k)} - \theta_{i-1}^{(k)}$.

Moreover, it is to be expected that for the conformal map f itself the error maxima will again occur at the boundary. So once the calculation of f is successfully done, it is useful to have a look at the expression $R(\arg f(e^{i\varphi})) - |f(e^{i\varphi})|$ for an additional error estimate.

In the present cases, all boundary curves are smooth enough to allow an application of WEGMANN'S method. Here, our implementation code mentioned on page 94 is quite optimized with respect to execution time by making use of symmetries, FFT and external provisioning of multiple used quantities. As in all smooth cases, using trigonometric approximation for the conjugate operator is best choice, the result for $\theta(\varphi)$ on geometry 3 using HERMITE interpolation for $R(\theta)$ is shown in Figure 7.

When using only 1024 discretization points on the unit circle, Figure 9(t) already shows a superb discrete error behaviour but this is only half of the rent. Recall that we are primarily interested in mapping the square to the geometry and when pre-joining the map from the disk with the closed-form SCHWARZ-CHRISTOFFEL transformation, our high-accurate discrete points on the disk will not be hit by the map coming from the

square. It is therefore necessary to focus on the continuous overall errors occurring on $[0, 2\pi]$. Depending on the geometry, their maxima turn out to be in range of $10^{-11}/10^{-6}/10^{-4}$ which is quite unsatisfactory. This can be improved by using 4096 discretization points. Results of the overall errors are shown in Figure 9(b).

	GEO1	GEO2	GEO3 (Hermite)		
θ	$\rho_1(\theta)$	$\rho_2(\theta)$	$\rho_3(\theta)$	$\rho'_3(\theta)$	$\rho''_3(\theta)$
0	7	6	5	5	20
$\pi/8$	9	9	—	—	—
$2\pi/8$	9	$7\sqrt{2}$	$7\sqrt{2}$	0	-30
$3\pi/8$	$15/2$	$15/2$	—	—	—
$4\pi/8$	$13/2$	7	7	0	25
$5\pi/8$	$15/2$	$15/2$	—	—	—
$6\pi/8$	9	$7\sqrt{2}$	$7\sqrt{2}$	0	-30
$7\pi/8$	9	9	—	—	—
π	7	6	5	-5	20
$9\pi/8$	$11/2$	5	—	—	—
$10\pi/8$	6	5	$7\sqrt{2}/2$	3	5
$11\pi/8$	8	$15/2$	$10\sqrt{2}/2$	10	30
$12\pi/8$	10	10	10	0	-50
$13\pi/8$	8	$15/2$	$10\sqrt{2}/2$	-10	30
$14\pi/8$	6	5	$7\sqrt{2}/2$	-3	5
$15\pi/8$	$11/2$	5	—	—	—
2π	—	—	5	5	20

Table 1: Definition of boundary points for trigonometric interpolation and derivatives (Hermite).

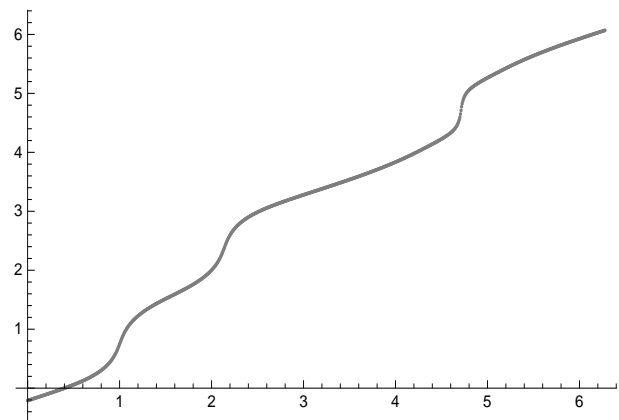


Figure 7: Function of boundary correspondence on GEO3.

¹basis=Union[Table[Cos[2m*phi], {m, 0, 4}], Table[Sin[(2m-1)*phi], {m, 1, 4}];
 lm=LinearModelFit[geo, basis, phi, WorkingPrecision->nprec, IncludeConstantBasis->False];
 R[phi_]=SetPrecision[lm["BestFit"], WorkingPrecision->nprec], Infinity];
²rfloat=Interpolation[geo, InterpolationOrder->3, PeriodicInterpolation->True, Method->"Hermite"];
 R[phi_]=SetPrecision[rfloat[phi], Infinity];

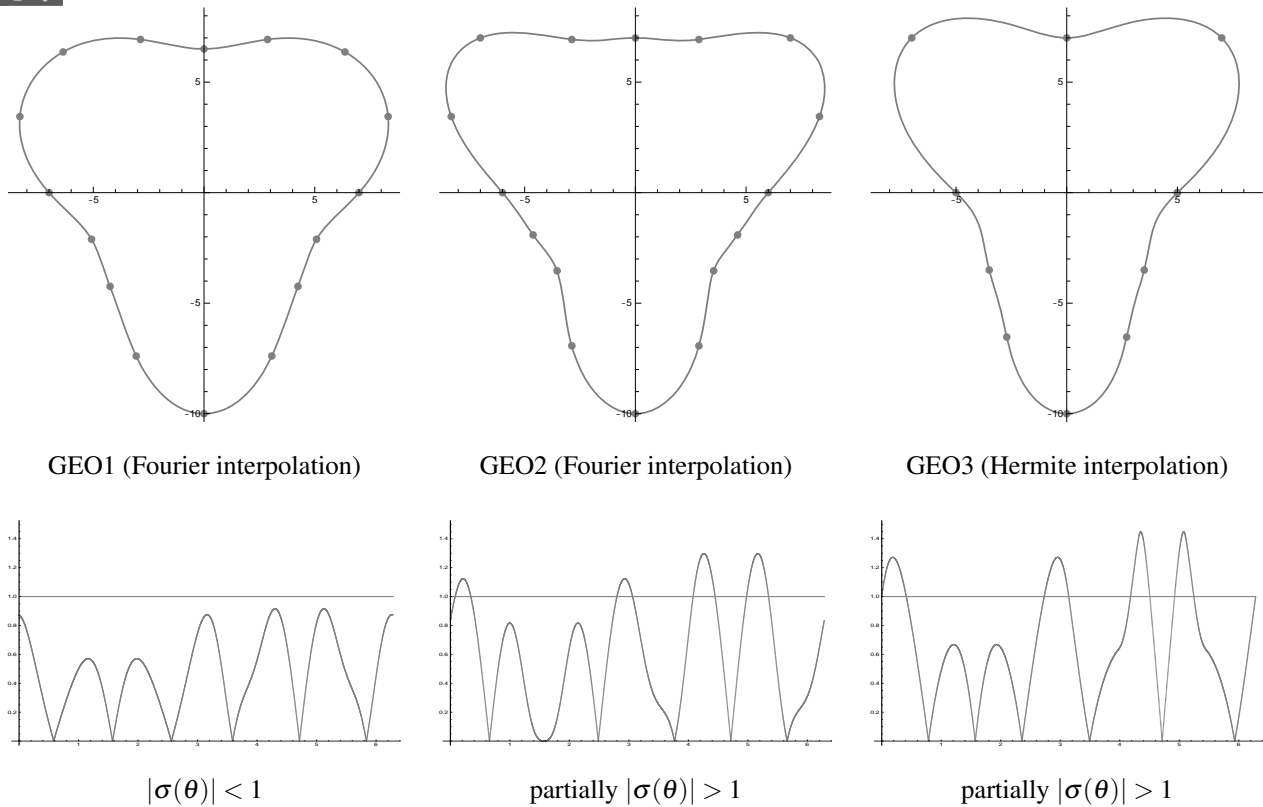


Figure 8: Points and interpolating functions $R(\theta)$ for the boundary curves (t). Symmetric $\rho(\theta)$ yields skew symmetric $\sigma(\theta)$ in $[0, \pi]$ and $[\pi, 2\pi]$ in respect of $\pi/2$ and $3\pi/2$ (b). The red line (symbolizing $\varepsilon = 1$) is used to investigate the ε -condition.

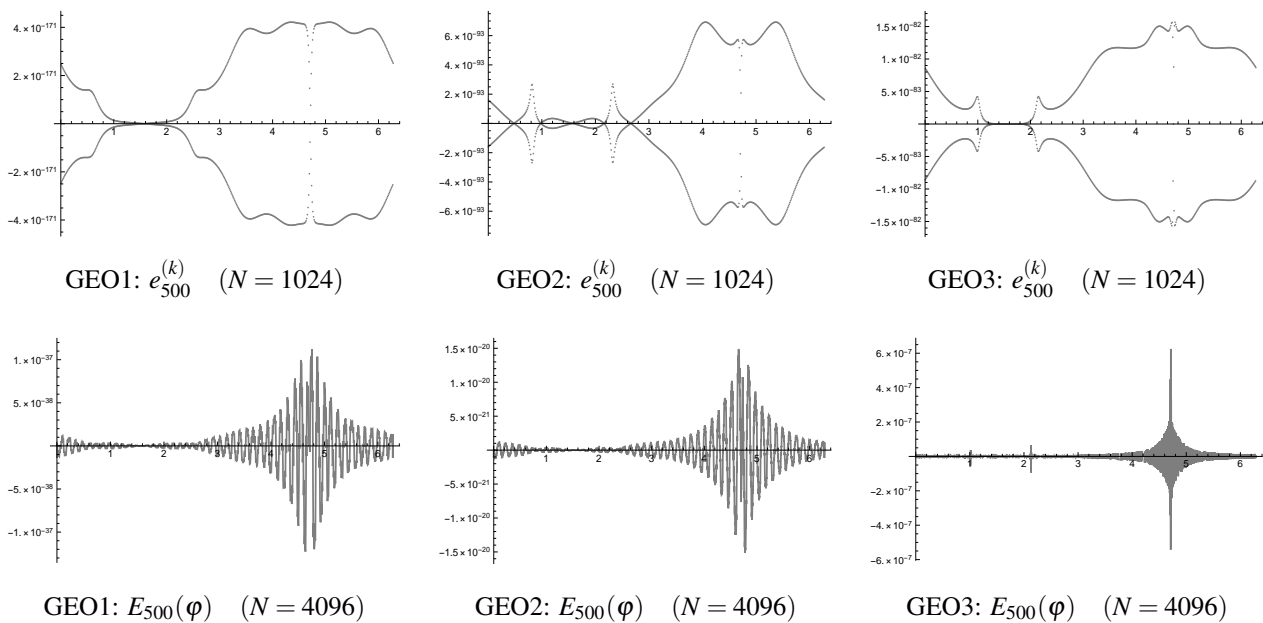


Figure 9: Discrete (t) and continuous (b) errors when iterating the function of boundary correspondence.

5 Conformal Co-ordinates

5.1 Metrics of the map

A given domain \mathfrak{G} in the complex plane can also be considered as a Riemannian manifold as a set of points. We are quite free how to address such a point $P \in \mathfrak{G}$, the most common way is the usage of Cartesian co-ordinates which can be expressed in terms of a homeomorphism $\kappa_\zeta : \mathfrak{D}_1 \subset \mathbb{R}^2 \rightarrow \mathfrak{G}$, hence $P = \kappa_\zeta(\xi, \eta)$ with the common Cartesian co-ordinates $\zeta = (\xi, \eta)$ of P . The two, here constant base vectors ∂_1, ∂_2 of the tangent space and the inner product of the manifold define

$$g_{ij}(P) := (\partial_i, \partial_j)(P) \equiv \delta_{ij}$$

as the metric (or fundamental) tensor. This can be our steppingstone to conformal parametrization.

Suppose now an other function $\kappa_z : \mathfrak{D}_2 \subset \mathbb{R}^2 \rightarrow \mathfrak{G}$ such that the point $P = \kappa_z(x, y)$ is assigned arbitrary co-ordinates. Then $z = (\kappa_z^{-1} \circ \kappa_\zeta)(\xi, \eta)$ acts as a transformation of co-ordinates (as far as a diffeomorphism). Let now $f : \mathfrak{D} \rightarrow \mathfrak{E} \rightarrow \mathfrak{G}$ be the conformal map considered before, then we can identify the way $f^{-1} = \kappa_z^{-1} \circ \kappa_\zeta$ performs the transformation of co-ordinates. The remaining thing to do is clarifying the new shape of the fundamental tensor and other metric quantities. With

$$\bar{\partial}_i = \frac{\partial x_j}{\partial \bar{x}_i} \partial_j \quad \text{and} \quad \bar{g}_{ij}(P) = \frac{\partial x_k}{\partial \bar{x}_i} \frac{\partial x_l}{\partial \bar{x}_j} g_{kl}(P)$$

as two cogredient transformation laws it is quite easy to derive $\bar{g}_{11} = \bar{g}_{22} = \xi_x^2 + \eta_x^2 = |f'(z)|^2 = 1/|f^{-1}(\zeta)'|^2$ and $\bar{g}_{12} = \bar{g}_{21} \equiv 0$. Thus our base vectors are not constant anymore but whirling around from point to point, being stretched by the same factor $\sqrt{\bar{g}_{11}}$ (we suppress the bar for convenience) but keeping orthogonality.

What becomes evident now is, that with the (numeric) evaluation of $f'(z) = \xi_x + i\eta_x$ we can calculate the metric tensor in each grid point of the unit square and save it for further disposal. Same holds for the second (and higher order) derivatives which can easily be calculated and saved for a grid via $f''(z) = \xi_{xx} + i\eta_{xx}$.

One reason for focusing on second derivatives is the importance of second kind CHRISTOFFEL symbols [2]. With the setting $\chi(x, y) := \xi_x^2 + \eta_x^2$ one concludes

$$\Gamma_{11}^1 = \Gamma_{12}^2 = \Gamma_{21}^2 = \frac{\xi_x \xi_{xx} + \eta_x \eta_{xx}}{\chi},$$

and the rest of them following likewise, revealing only two of eight independent expressions [5, p.50ff.].

5.2 Differential operators

For the development of a simulation code involving PDEs it is now essential to formulate the underlying laws in co-ordinate free manner and then realize them in conformal co-ordinates, making use of the metrics. To be confusing, covariant differentiation of a tensor field can be used in such formulations: 'ordinary' partial derivatives which we treated with spatial approximation formulae applying Finite Difference schemes must be corrected by sums involving CHRISTOFFEL quantities, depending on their co- or contravariant nature we get

$$\frac{\partial \Phi_{j_1 \dots j_m}^{i_1 \dots i_n}}{\partial x_p} = \frac{\partial \Phi_{j_1 \dots j_m}^{i_1 \dots i_n}}{\partial x_p} + \sum_{h=1}^n \Gamma_{lp}^{i_h} \Phi_{j_1 \dots j_m}^{i_1 \dots i_{h-1} i_{h+1} \dots i_n} - \sum_{k=1}^m \Gamma_{kp}^{j_k} \Phi_{j_1 \dots j_{k-1} j_{k+1} \dots j_m}^{i_1 \dots i_n}$$

and as an example, the LAPLACE operator for a scalar field in arbitrary co-ordinates will then be written³ as

$$\Delta u = g^{ij} \frac{\partial^2 u}{\partial x_i \partial x_j} + \left(\frac{\partial g^{ij}}{\partial x_j} + \frac{g^{ij}}{2g} \frac{\partial g}{\partial x_j} \right) \frac{\partial u}{\partial x_i}.$$

This means a quite nice generalization of the included Cartesian case, where $g_{ij} = \delta_{ij}$ yields the well-known expression involved in the heat equation. Moreover, inserting the derivative f' and doing a bit of calculation legitimates the replacement $\Delta u \rightarrow \Delta u / (\xi_x^2 + \eta_x^2)$ we used in our previous article (5, heat equation on disk).

We conclude that conformal parametrization of a given domain by the unit square results in a minimum of overhead with respect to curvilinear co-ordinates.

5.3 Boundary conditions

While DIRICHLET boundary conditions pose no further problems as the values of a state variable are prescribed, NEUMANN/NEWTON type conditions require the transformation of the outward pointing normal derivative. Here another benefit of the conformal map appears, the direction of such a vector always coincides with one of the local base vectors. Respecting orientation, we have only to correct its length and derive [5, p.53]

$$\nabla_n u = \pm \frac{1}{\sqrt{\xi_x^2 + \eta_x^2}} \frac{\partial u}{\partial x_i}.$$

With these results we are able to simulate time-dependent PDEs where state variables can even be tensor-valued. Do not forget to transform them by using the correct transformation laws in your implementation.

³Einstein summation, $g^{ik} g_{kj} = \delta_j^i$ and g denotes the determinant of g_{ij} .

Summary

In this contribution we focused on the numerical solution of THEODORSEN's nonlinear and singular integral equation to establish the function of boundary correspondence, $\theta(\varphi)$. With pre-given $\rho(\theta)$ the conformal map $f: \mathfrak{E} \rightarrow \mathfrak{G}$ can be constructed either by NEWTON iteration using Fourier expansion of $\ln \rho(\theta(\varphi))$ or - in case of increasing ε or vertex singularities - by means of fix point iteration, direct evaluation of the involved nonlinear and singular integral with successive application of SCHWARZ or CAUCHY formulae.

Neither way poses problems for an implementation in *Mathematica* and with the error estimates for $\theta(\varphi)$ on the boundary we control the overall error on the disk.

If trigonometric interpolation turns out to be too inaccurate as in case of present vertices (we considered the unit square), then the corner singularities have to be investigated and usage of more apt base functions can ship around the problem to find a more accurate approximation for the function of boundary correspondence.

We note that THEODORSEN's equation is only one of plenty approaches to numerically construct the map. If results are unsatisfactory concerning accuracy then heading out for other integral equations or approximation methods provided in literature is an idea.

In a bigger context, the obtained conformal map (and its derivatives) can be utilized in PDE-based simulations. This approach is convenient to minimize the complexity involved when introducing curvilinear coordinates. Moreover, starting with a co-ordinate free description, one is able to formulate the laws regarding metrics of the concrete conformal transformation.

From the co-ordinate point of view, the conformal metric of course alters the shape of the PDE and after all, co-ordinates are necessary for doing number crunching simulations while the underlying physical laws indeed stay invariant. We shall present such a practical simulation study in a subsequent contribution.

With this in mind, a numerical treatment of PDEs by using conformal maps combined with the CTDS method of lines gets less a matter of consistency than rather an issue of stability: let us therefore interpret the approximation error committed by the numerical determination of the function of boundary correspondence as a slight perturbation of the metric quantities. The smaller this error can be squeezed, the better will be the quality of out-coming results of our overall simulation.

References

- [1] Blümlinger, M. *Konforme Abbildungen von Gebieten mit Ecken*. TU Wien, Dipl.-Arb., 1984.
<http://katalog.ub.tuwien.ac.at/AC00227190>
- [2] Dirschmid, H.J. *Tensoren und Felder*. Springer-Verlag, Wien, 1996, ISBN 3-211-82754-4.
- [3] Gaier, D. *Konstruktive Methoden der konformen Abbildung*. Springer-Verlag, Berlin-Göttingen-Heidelberg, 1964.
- [4] Holzinger, M. *CTDS – Mauling Heat Equation on Unit Disk by Conformal Parametrization*. Simulation Notes Europe 2020; 30(2): 43-50.
DOI: 10.11128/sne.30.tn.10511.
- [5] Holzinger, M. *Konforme Abbildungen zur Simulation von Modellen mit verteilten Parametern*. Dissertation/PhD (in German), Wien, 2020.
<http://katalog.ub.tuwien.ac.at/AC15652638>
- [6] Hübner, O. *The Newton method for solving the Theodorsen integral equation*. Journal of Computational and Applied Mathematics 14 (1986), pp. 19-30.
- [7] Kythe, P.K. *Computational Conformal Mapping*. Birkhäuser, Boston 1998, ISBN 0-8176-3996-9.
- [8] Richardson, S. *Integral Equations*. Mathematica Journal, Vol. 9, Issue 2, pp. 460-482, 2004.
<https://www.mathematica-journal.com/issue/v9i2/contents/IntegralEquations/IntegralEquations.pdf>
- [9] Wegmann, R. *Discretized versions of Newton type iterative methods for conformal mapping*. Journal of Computational and Applied Mathematics 14 (1986), pp. 207-224.

```

(* Discrete Newton Iteration According To Hübner/Wegmann;  $\eta$ -axis symmetry *)
Clear["Global`*"]; WD=FileNameTake[NotebookDirectory[],5];SetDirectory[WD];
nprec=nprecfourier=500; znp=4096; np=znp/2; niter=500;
(* load external geometry definition to make use of r[phi_] *)
Import[FileNameJoin[WD,"GLOBAL","Geo2_Definition_Fourierpoly.wl"]]
(* initialize phi_k und start-vector *)
phicore=Table[N[x,nprec],x,Pi/2,3Pi/2,Pi/np];
phia11=Table[N[x,nprec],x,0,2Pi-Pi/np,Pi/np]; svalues[i_]=phia11;
(* some terms can be calculated ex ante *)
sin2m = Chop[Table[Sin[2*m*phicore],m,1,np/2],10^-nprec]; (* rest analog *)
(* trigonometric interpolation polynomial *)
pt[t_,m_,y_]:= 1/Sqrt[Dimensions[y][[1]]] * ( Re[y[[1]]]+
  2*Sum[(Re[y[[k+1]]]Cos[k*t]-Im[y[[k+1]]]Sin[k*t]),{k,1,(m/2)-1}] +
  Re[y[[1 + Dimensions[y][[1]]/2]] Cos[(m/2)*t] );
(* make use of symmetries: calculate on [Pi/2,3Pi/2] extend to [0,2Pi] *)
oddeextension[vec_]:=Flatten[Append[Prepend[vec,Table[-vec[[np/2+1-k]],
  {k,0,np/2-1}]],Table[-vec[[np-k]],{k,0,np/2-2}]]];
evenextension[vec_]:=Flatten[Append[Prepend[vec,Table[vec[[np/2+1-k]],
  {k,0,np/2-1}]],Table[vec[[np-k]],{k,0,np/2-2}]]];
(* begin iteration loop *)
Do[ t0 = Timing[
lnRvalues=Log[r[svalues[i-1]]];sigmavalues=r'[svalues[i-1]]/r[svalues[i-1]];
Block[{$MinPrecision=nprecfourier,$MaxPrecision=nprecfourier},
  fklr=Fourier[lnRvalues,FourierParameters->{0,-1}]];
lnRf[phi_] = Expand[pt[phi,Dimensions[fklr][[1]],fklr]];
anlnR=Append[Coefficient[lnRf[phi],Table[Cos[2*m*phi],{m,1,np/2-1}]],0];
bnlnR=Coefficient[lnRf[phi],Table[Sin[(2m-1)phi],{m,1,np/2}]];
(* calculate K[lnR]/K[v]/K[q] on [Pi/2,3Pi/2] for efficiency sake *)
klnRcore=Chop[Total[anlnR*sin2m-bnlnR*cos2mm1],10^-nprec];
klnRvalues=oddeextension[klnRcore]; fvalues=svalues[i-1]-phia11-klnRvalues;
vvals=ArcTan[sigmavalues]; rvalues=Sqrt[1+sigmavalues^2];
Block[{$MinPrecision=nprecfourier,$MaxPrecision=nprecfourier},
  fkv=Fourier[vvals,FourierParameters->{0,-1}]];
vfourier[phi_] = Expand[pt[phi,Dimensions[fkv][[1]],fkv]];
anv=Coefficient[vfourier[phi],Table[Cos[(2*m-1)phi],{m,1,np/2}]];
bnv=Append[Coefficient[vfourier[phi],Table[Sin[2*m*phi],{m,1,np/2-1}]],0];
w=Chop[Total[anv*sin2mm1-bnv*cos2m],10^-nprec];wvalues=evenextension[w];
alpha=Chop[(1/znp)*Sum[vvals[[1]],{1,Dimensions[vvals][[1]]}],10^-nprec];
q=(fvalues*sigmavalues)/(rvalues*Exp[wvalues]);
Block[{$MinPrecision=nprecfourier,$MaxPrecision=nprecfourier},
  fkg=Fourier[qvalues,FourierParameters->{0,-1}]];
qfourier[phi_] = Expand[pt[phi,Dimensions[fkg][[1]],fkg]];
anq=Append[Coefficient[qfourier[phi],Table[Cos[2*m*phi],{m,1,np/2-1}]],0];
bnq=Coefficient[qfourier[phi],Table[Sin[(2*m-1)*phi],{m,1,np/2}]];
pcore=Chop[Total[anq*sin2m-bnq*cos2mm1],10^-nprec];p=oddeextension[pcore];
qdach=Chop[(1/znp)*Sum[q[[1]],{1,Dimensions[q][[1]]}],10^-nprec];
svalues[i]=svalues[i-1]-(fvalues/(rvalues^2)) -
  (((p+qdach*Tan[alpha])*Exp[wvalues])/rvalues);];
t0 // Print, {i, 1, niter} ]; (* iteration step time, end iteration loop *)

```

Generating Logistic Characteristic Curves using Discrete Event Simulation and Response Surface Models

Sonja Kuhnt¹, Dominik Kirchhoff¹, Sigrid Wenzel^{2*}, Jana Stolipin²

¹Department of Computer Science, FH Dortmund University of Applied Sciences and Arts, Emil-Figge-Str. 42, 44227 Dortmund, Germany;

²Department of Production Organization and Factory Planning, University of Kassel, Kurt-Wolters-Str. 3, 34125 Kassel, Germany; *s.wenzel@uni-kassel.de

SNE 30(3), 2020, 95-104, DOI: 10.11128/sne.30.tn.10522
Received: August 4, 2020; Revised: August 15, 2020;
Accepted: August 20, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Logistic Characteristic Curves (LCCs) or Logistic Operating Curves (LOCs) describe relationships between various Key Performance Indicators (KPIs) of production and logistics systems. These relationships can be qualitatively or quantitatively visualized by charts to illustrate the performance of these systems. Discrete Event Simulation (DES) allows a detailed investigation of the dynamic behavior of production and logistics systems under consideration of uncertainties and thus contributes to their planning reliability. Using simulation models and the data generated by the experiments, KPIs of the modeled systems are measured. Of course, different production and logistics systems also have several target systems whereby the individual target variables interact with each other and can, therefore, conflict. In this paper, a methodology is presented that combines DES and a statistical technique for empirical model building, namely the response surface model, to predict the behavior of production and logistics systems by using LOCs and thereby decrease the effort for experimentation by reducing the number of simulation runs.

Introduction

Logistic Characteristic Curves (LCCs) graphically show the qualitative and quantitative interrelationships between logistic key performance indicators (KPIs) in an in-house production and logistics systems.

Often, different logistic KPIs are in a field of conflict between logistic performance and logistic costs, and it is impossible to achieve a system optimum for all of them at the same time. Generally, the goals given for a planning task must be positioned within the logistic KPIs. LCCs, also known as Logistic Operating Curves (LOCs), provide a tool for supporting this process [1, 2]. In difference to performance indicators, which can also be measured in a running production process, complete operating curves can only be determined by mathematical or simulation models, since experiments on the real system are not possible for economic reasons.

In the field of production and logistics system planning, Discrete Event Simulation (DES) is a recognized problem-solving method [3] and, therefore, also an important Digital Factory method [4]. DES supports the detailed examination of production and logistics systems with discrete unit loads by analyzing the dynamic and random behavior to uncover possible planning errors and inaccuracies in behavior. Simulation offers the possibility to examine a system without developing or disturbing real systems, so that process alternatives can be checked, system structures can be varied, and different controls and parameter settings of systems can be tested. The better the data situation of a simulation study is (i.e., the output data of the experiments), the more precise are the statements about the system that are to be made based on the designed experiments. Both the design of suitable experiments and the analysis of the data generated by the simulation models are challenges in a simulation study.

In this paper, we propose to derive LCCs for a simulation model of a manufacturing logistics system by statistical metamodels.

Thereby LCCs are able to be inferred for complex systems. It is also possible to show the effect of several system parameter configurations at the same time. The general idea behind dealing with expensive function evaluations or simulation experiments by using surrogate- or metamodels is discussed in Section 1, which also deals with the concept of LCCs and DES. In Section 2, we introduce the general approach to derive LCCs from a limited number of well-selected simulation runs by building metamodels for selected logistic KPIs. The approach is exemplified in Section 3 by an application to a manufacturing logistics system with two workstations, as described in the DOSIMIS-3 tutorial [5]. We conclude with a summary and outlook in Section 4.

1 Background

1.1 Logistic characteristic curves

Using LCCs, the interrelationships between logistic KPIs in in-house production and logistics systems are shown in the form of graphs. The core processes of these systems comprise procurement, production, and distribution [1, 2]. The LOC theory [6] is built on a deductive-experiment effects model. This model is used as a starting point for the development of approximation equations for the mathematical description of a real logistics system using LOCs. These curves reduce the complexity and, therefore, the costs of modeling the behavior of logistics systems [8]. With the help of LOCs, it is possible to describe KPIs of systems. The principle of mapping the relationships between the effects of the KPIs of production and logistics systems using the LOC theory can be transferred to the production areas for each networked system. LOCs can be used to control and improve the reliability of logistical system performance [1]. LOCs realize effectivity support in the planning and control of operations by giving qualitative and quantitative descriptions of, for example, the KPI Work-In-Process (WIP) and its effects on the output rate and the throughput time. The objectives can be weighted, and then the parameters can be adjusted. An enterprise can adapt its internal requirements in conformity with the market. LOCs make it possible to understand and mathematically describe the interrelationships between the logistic goals such as output rate, lead-time, and WIP [8]. They are an excellent basis for supporting and monitoring the process safety and capability and can be used to evaluate the

process during production control. For instance, under the existing structural condition, they can demonstrate that throughput times and WIP value can be reached at one workstation without having to calculate the material flow disruptions or the associated power losses. For a successful implementation of LOCs in production planning and control, goal-oriented system parameters are derived and adapted [7, 8]. The LOC theory is closely related to the simulation of production and logistics systems. The evaluation of the system performance is a central issue in production planning and control, as well as in simulation and optimization of production and logistics processes. Often, specific parameters are used, which only allow an evaluation in the selected operating point. Unlike the performance parameters, which can also be measured in a running production process, complete operating curves can only be determined through simulation models, since experiments in the real system are not possible for economic reasons. Real logistics systems can be compared with each other using the standardized characteristic curves. In most cases, however, the reference values for this standardization can only be determined using simulation methods [9].

1.2 Discrete event simulation

The use of the DES in production and logistics is presented in detail in the series of standards VDI 3633 [10], as well as in [11], [12], and others. DES differs from continuous simulation methods by mapping the progress of time via atomic events that trigger state changes. Each event has a timestamp and is added to an event queue, the simulation time is set according to the timestamp (from the first event in the queue), and then the event is processed [3]. The scope of DES applications ranges from the consideration of in-house manufacturing logistics to procurement and distribution logistics. The evaluation criteria in a simulation study are determined according to the objectives and the object of investigation. Among other things, buffer dimensions and performance data of the machines are evaluated and processing times, transport speeds, or batch sizes are varied in order to determine throughput times and capacity utilization or to reduce transport, or storage times. A simulation-supported examination can prove, for example, whether the planned production quantity can be handled, transported, and produced based on the production concept developed. Within the framework of a simulation study, a model is built using a suitable software tool.

The simulation models are used in the operating phase, as well as in the planning phase of production and logistics systems. By means of a systematic variation of individual parameters of a simulation model and corresponding experiments, planning concepts can be analyzed according to the defined objectives of a simulation study. To be able to make more precise statements about the systems under investigation, it is necessary to examine a large volume of simulation data. When investigating complex production and logistics systems with comprehensive questions, not only valid models have to be built, but also many complex experiments with a lot of simulation runs must be carried out. The design and implementation of experiments are usually very complicated. However, the generated experimental output data is not the only result of a simulation study. A visualization of the input and output data of the simulation model supports the preparation of the simulation results. The interpretation of the experimental data can be reasonably supported by the methods of statistics and data visualization.

1.3 Metamodel-based analysis – response surface models

The word “metamodel” points to the fact that this technique is used to generate a model of a model. A metamodel, or surrogate model, approximates the relationship between the varied input parameters, also called input variables, of a simulation model and so-called response or target variables using a limited number of well-chosen points generated with the simulation model [13]. These points, i.e., settings of the input variables, are determined by a statistical experimental design, which depends on the kind of metamodel and the problem to be solved. Metamodels can be either interpolating or smoothing. The so-called Kriging model is a popular interpolating surrogate model and is most often used for solving expensive black-box optimization problems [13]. It has the ability to not only give point predictions on the target variable, but also to provide the predictive distribution and thereby an uncertainty evaluation over the complete space of input variables. Response surface models, on the other hand, are a class of smoothing models [14]. They can be used for solving optimization problems as well but are especially suitable for the approximation of the unknown relationship between input variables and response variables – the so-called response surface – with minimal quadratic loss.

Given a set of independent input variables x_1, \dots, x_p , the model has the form $Y = f(x_1, \dots, x_p) + \varepsilon$, where Y denotes the response considered, i.e., in our context one of the logistic KPIs, f is typically a low-order polynomial and ε a Gaussian error term. Predictions of the expected response achieved by least-square estimates of the unknown parameters of the function f , provide LCCs. The accuracy of a fitted response surface model may be evaluated by some additional simulation runs and observing the differences between the outcomes and the predictions given by the model. The finally achieved LCCs not only capture the behavior of the logistic KPIs over the whole domain of different input variables but, moreover, the uncertainty inherent to LCCs can be presented by prediction intervals.

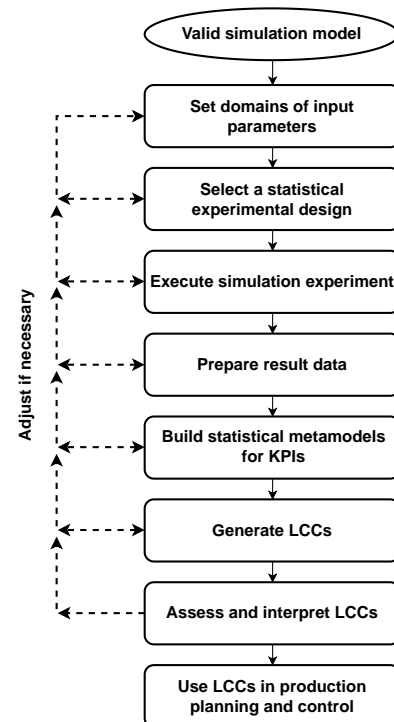


Figure 1: Procedure model for the metamodel-based generation of LCCs

2 LCCs from metamodeling of a simulation model

We now use the metamodel-based analysis of a simulation model to derive LCCs. Our approach is summarized in Figure 1. As a starting point, we assume the existence of a valid simulation model. It must present a reliable image of the real system to be analyzed and return the logistic KPIs of interest as output. We take it that the typical simulation model used in the context of logistics will have some element of uncertainty, e.g., by system loads generated by some statistical distribution. Simulation runs with the same parameter setting but different seeds of the involved random number generator, i.e., replications, then vary in their output. We, therefore, concentrate on response surface models as metamodels in our application in Section 3.

Before we can build a metamodel, we need to discuss which input variables are assumed or known to have an effect on the KPIs and in which range or on which values they can be varied. If we have a list of potential input variables to start with, we might have to conduct a screening analysis first. This can be done by fitting a first-order main effect regression model to the result of a limited number of simulation runs. The primary goal is a rough understanding of the process and to identify the least influential variables by using as few simulation runs as possible. If some of the candidate variables do not display an apparent effect on the measured KPIs, they are set to a fixed value. For the remaining input variables, domains of interest need to be determined, or more specifically, it has to be decided which values or intervals of them are later to be displayed in the LCCs.

Next, a statistical experimental design is set up, which gives the settings of the simulation runs to be done. A more flexible model like a second-order polynomial response surface model with interactions is then to be built from the results of these runs as the basis for the LCCs. Typically, for Kriging models, the methods of choice are so-called Latin Hypercube Designs as they cover the whole domain and thus are called space-filling designs. Many different experimental designs exist for response surface models. The most popular are two-level factorial and two-level fractional factorial designs for variable screening – i.e., finding out which input variables have an effect on the response and which are negligible.

For the actual response surface fit, often central composite designs, D-optimal designs, or even higher-order factorial designs are the methods of choice [14].

All input variable settings contained in the selected statistical experimental design are then executed. As a result, we have a data set containing measured values for the KPIs for each executed simulation run.

This set of data is subject to an initial analysis to detect possible inconsistencies and get a first impression of structural features. At this point, it might be necessary to return to the beginning of the procedure model (see Figure 1), e.g., if unforeseen effects within the simulation model require an adaption of it.

The next goal is to build a metamodel from the simulation runs, which approximates the true response surface well. A well-chosen metamodel reflects the influences of input parameters and their interdependencies on the KPIs.

LCCs are then created by graphically displaying prediction curves from the response surface (or Kriging) model. To see the effect of different logistic strategies and uses of resources on the logistic KPIs, the predictions are displayed against the varied input variables. Both, response surface models and Kriging models, provide an uncertainty quantification of the prediction, which can be presented as valuable additional information on the plot. While assessing and interpreting the derived LCCs, issues might occur which make a return to previous steps of the procedure model and an adjustment of them necessary, for example, by extending or reducing the range of the input variables. Finally, the procedure model delivers LCCs to be used in planning and control of production and logistics systems.

3 Demonstration of the procedure model

3.1 DOSIMIS-3 simulation model

For the demonstration of the procedure model described above, the finished basic model of the DOSIMIS-3 tutorial (Figure 2) is used. DOSIMIS-3 is a simulation tool that allows the graphical modeling, visualization, simulation, and analysis of production and logistics systems. The simulation model is described in detail in the DOSIMIS-3 user manual of the SimulationsDienstleistungsZentrum GmbH [5] and is one part of the corresponding DOSIMIS-3 tutorial.

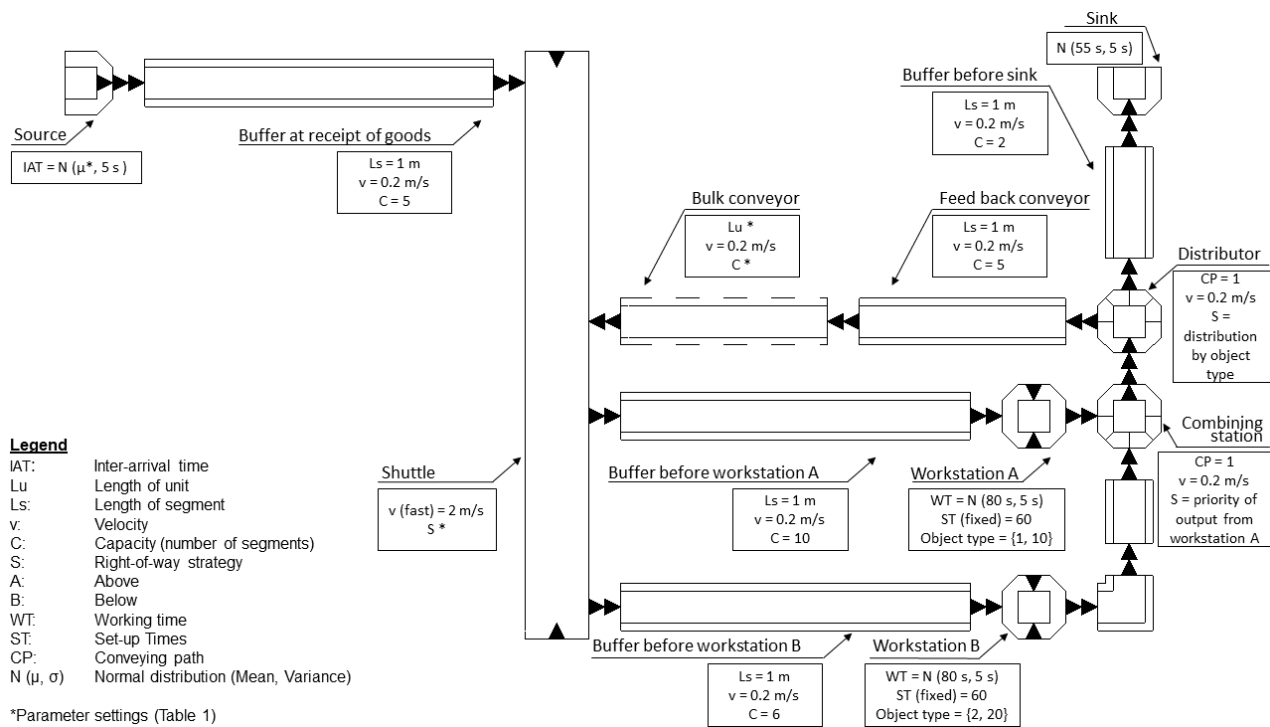


Figure 2: Simulation model in DOSIMIS-3 (for the basic model see [5]).

Model element	Parameter	Range
Source	system load	[54, 62]
Shuttle	right-of-way strategy	{FIFO, Priority to correction route (Prio Corr.)}
Bulk conveyor	batch size	{1, ..., 8}

Table 1: Input variables and selected parameter settings.

The analyzed logistics system is an example of a production process from the electricity industry, which manufactures consumer goods. The borders of this system are the receipt of parts (one source and one queue) and the exit of goods (one queue and one sink). The two product parts (A and B) are randomly generated by the source and conveyed via a shuttle to one of the two workstations (type A to the upper one and type B to the lower one) since each workstation can only produce one product type. In the simulation model, product part type A is coded with 1 and product part type B with 2. The quality control of the products is performed in a separate section, as can be seen in Figure 2. Those products that have some defect must return to one of the two workstations, depending on the type of product, via a shuttle based on the disposition of batch lots.

The faulty product parts are marked either with 10 (for type A) or with 20 (for type B). Those that have no defects leave the model at the sink. At the workstations, a set-up is necessary between the parts coming from the source and the faulty parts. As a consequence, there are high set-up times in the model [5].

The model used in our example is improved by comparison to the basic model of the tutorial (the parameter settings are shown in Figure 2). Therefore, the velocity (v) of the shuttle, as well as the buffer sizes, are improved, and a bulk conveyor organizes different batch sizes.

The simulation model is used to check the effects of different *right-of-way strategies* of the shuttle (“FIFO”: First In First Out, or “Prio Corr.”: prioritize correction route), as well as *system loads* at the source, and dif-

ferent *batch sizes* on the possibility to reach a desired throughput per hour. In addition, the simulation model is used to examine the *utilization rate* of the two workstations and to uncover possible bottle-necks in the system.

3.2 Response surface model-based LCCs

Based on the described DOSIMIS-3 simulation model, we apply the procedure model from Section 2 to derive LCCs for the KPIs *average throughput* and *utilization rate* at the upper workstation. The KPI average throughput is the average number of parts processed per hour. The KPI utilization rate is measured as the time the workstation is in use in percentage, including set-up times. Each of our simulation runs covers five days of simulation time with a four hour lead time. As an advantage of our metamodel-based approach, we can explore the behavior of these KPIs depending on various input variables, here the *right-of-way strategy*, the *system load*, and the *batch size*.

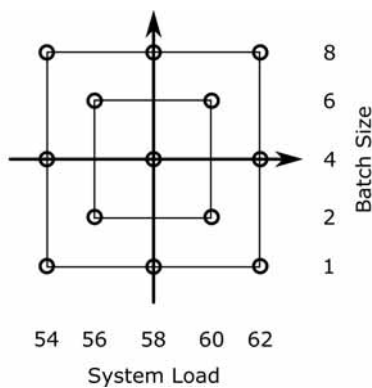


Figure 3: Settings of *system load* and *batch size* used in the statistical experimental design.

Table 1 contains the possible values these input variables can be set on in the simulation model. As the *system load* may be set on any value in the considered range from 54 to 62, we get an infinite number of possible settings. Now, consider reducing the choice to every integer between 54 and 62, i.e., nine *system loads*. Together with the eight possible *batch sizes* in $\{1, \dots, 8\}$ and the two considered *right-of-way strategies*, this leads to 144 possible combinations. As we plan to replicate each considered parameter setting three times with different random seeds, this would result in 432 simulation runs.

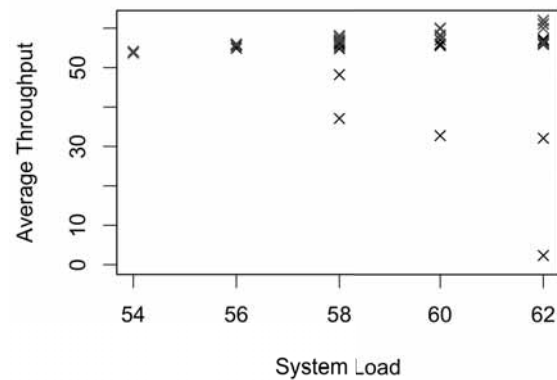


Figure 4: Scatterplot of *average throughput* vs. *system load*. The black points belong to the *right-of-way strategy* "FIFO" while the blue points belong to "Prio Corr.".

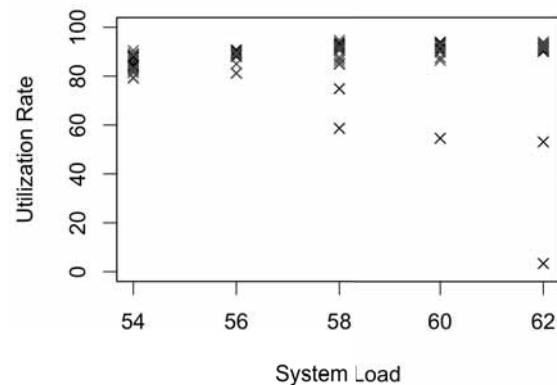


Figure 5: Scatterplot of *utilization rates* vs. *system loads*. The black points belong to *right-of-way strategy* "FIFO", the blue points to "Prio Corr.".

Instead, using the statistical Design of Experiments (DoE), we decide on a combination of a Central Composite Design (CCD) with axial points on the box and a full factorial design for two intermediate levels, which is to be conducted for each of the two *right-of-way strategies*. CCDs are classical design for the fitting of second-order response surface models [14]. A CCD consists of a full-factorial design of all combinations of high and low values (54 and 62 for *system load* and 1 and 8 for *batch size*), a center point (*system load* = 58 and *batch size* = 4 in our case) and so called axial or star points. In our chosen design, the axial points are on the box of the full-factorial design to cover the boundary area well.

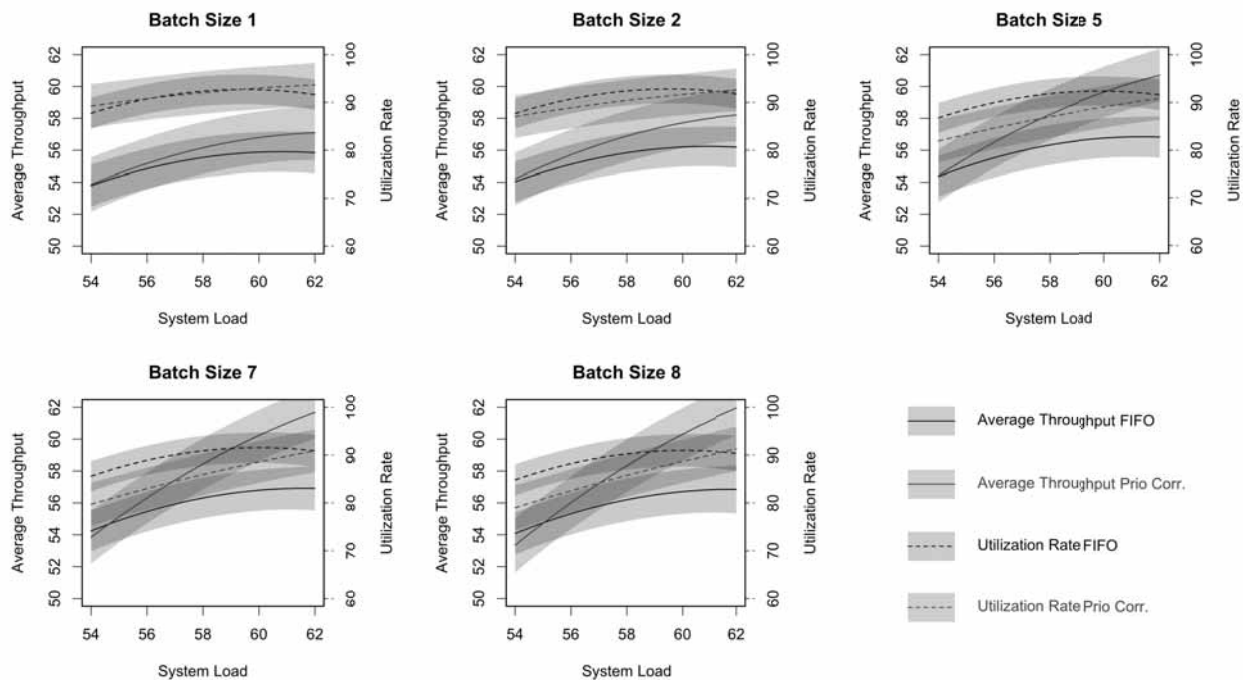


Figure 6: LCCs of average throughput and utilization rate of the upper workstation in dependence of system load, selected batch sizes, and both right-of-way strategies.

We extend the CCD by adding a full-factorial design on two medium levels (*system load* = 56/60 and *batch size* = 2/6) to achieve a more space-filling effect. Figure 3 indicates the 13 combinations of values for the *system load* and the *batch size*. These parameter settings are used for both *right-of-way strategies*, resulting in 26 different parameter settings in our experimental design. Each of these settings is replicated three times. Using a new random seed for each run, altogether 78 simulation runs are done, and the KPIs are observed.

Figure 4 and 5 show the *average throughputs*, and the *utilization rates* of the top workstation, respectively plotted against the *system load*. It is evident that most of the points' *average throughputs* are close to the related *system loads*. However, there are some deviating values with less than 50 parts per hour in average. For the *utilization rate*, a similar phenomenon can be observed. The outliers of both KPIs are identical and belong exclusively to the *right-of-way strategy* "FIFO". In these runs, a deadlock occurred. We omit the five runs in our building of LCCs by response surface models as they obviously display a very different pattern from the behavior of the remaining simulation results.

We build linear regression models for the KPIs by least-square estimation and separately for each *right-of-way strategy*. For the remaining input variables *system load* and *batch size*, a full second-order model of main effects, quadratic terms, and the interaction term is fitted [15]. The adjusted determination of coefficients (R^2) measures the goodness-of-fit of a regression model with possible values between 0 and 1. For the four fitted models, R^2 takes values between 0.84 and 0.95, indicating a good fit in each case.

Figure 6 contains LCCs for both *average throughput* and *utilization rate* of the upper workstation. They consist of the models' predictions (the lines) together with the 99% prediction intervals (the ribbons). Note, that we can generate these predictions for all parameter settings within the domain. We are especially not restricted to *batch sizes* 1, 2, 4, 6, and 8 only. Here, we display LCCs for small *batch sizes* (1 and 2), for a medium *batch size* (5) and for *batch sizes* at the upper end of our considered range (7 and 8).

For example, the solid line depicted in Figure 6 for the *average throughput* in case of “Prio Corr.” as *right-of-way strategy* is calculated by the estimated prediction equation

$$\begin{aligned} \text{average throughput}_{\text{Prio Corr.}} = & -105.766 \\ & + 5.345 \cdot \text{system load} - 4.612 \cdot \text{batch size} \\ & - 0.043 \cdot \text{system load}^2 - 0.070 \cdot \text{batch size}^2 \\ & + 0.096 \cdot \text{system load} \cdot \text{batch size}. \end{aligned}$$

It is obvious that the LCCs vary between *batch sizes* and *right-of-way strategies*. They show what can be achieved with the different considered variations of the simulated production process. For example, in case that a *system load* of 54 parts per hour has to be processed, the *average throughput* should also equal 54. This can be achieved with both *right-of-way strategies* using only a *batch size* of 1. However, for larger *system loads*, “FIFO” as a *right-of-way strategy* is not successful. With higher required *average throughputs*, “Prio Corr.” needs to be chosen along with increasing *batch sizes*. E.g., a *system load* of 58 requires at least *batch size* 5. With an even higher *batch size*, the *utilization rate* can be reduced. The upper limit of considered *system loads* can only successfully be processed by using at least *batch size* 8.

3.3 Validation of the procedure model

To validate our procedure model in the case of the DOSIMIS-3 simulation model, we conduct further simulation runs. We extend our data set to contain each combination of *system loads* with an integer value between 54 and 62, *batch sizes* 1, 2, 4, 6, and 8, as well as the two *right-of-way strategies*, replicated three times. Altogether, we end up with 270 simulation runs.

In Figure 7, we have plotted the data points against the model’s predictions for the *average throughput*. Within the additional simulation runs, seven more deadlocks for *right-of-way strategy* “FIFO” occurred, which we exclude for clarity’s sake.

We can see that only a few points do not lie within the prediction intervals, e.g., for “FIFO”, a *system load* of 61 and *batch sizes* of 4 or 6. The three replications for each of these two locations, exhibit a greater variance than most of the other replications, which cannot be covered by the model. Altogether, the model shows a good prediction accuracy.

Figure 8 shows the predictions of the *utilization rate* of the upper workstation for the same *system loads* along with the 99% prediction intervals. For the *right-of-way strategy* “Prio Corr.”, all of the points lie within the prediction interval. For “FIFO” again, a few points lie outside the interval. However, the overall behavior of the model seems more than plausible.

4 Summary and outlook

In this paper, we have presented a new procedure model to obtain LCCs based on simulation models and statistical metamodeling. Although only a few simulation runs are required, it is possible to describe the system behavior and the resulting KPIs depending on different system configurations. Using statistical models also allows for quantifying the system-inherent uncertainty. With a first example of application, the basic feasibility and usefulness of the procedure model are shown. Based on these results, other research is planned to prove the applicability and the general validity of the procedure model, also for more complex systems and further KPIs. Extensions of the approach and new methodological developments also appear to be useful with regard to statistical modeling. For example, the prediction intervals capturing the uncertainty are symmetrical in the presented simulation model, although, the *average throughput*, in particular, cannot, in general, be higher than the *system load*. Also, the occurring deadlocks are not pursued further in our investigations. It is conceivable to define the probability of the occurrence of deadlocks as a KPI through suitable metamodels and to represent it in LCCs.

References

- [1] Nyhuis P, Wiendahl, H-P. *Fundamentals of Production Logistics*. Springer: Heidelberg; 2009.
- [2] Hopp WJ, Spearman ML. *Factory physics. Foundations of manufacturing management*. Irwin/McGraw-Hill: Boston; 2001.
- [3] Gutenschwager K, Rabe M, Spieckermann S, Wenzel S. *Simulation in Produktion und Logistik. Grundlagen und Anwendungen*. Berlin: Springer Vieweg; 2017.
- [4] Bracht U, Geckler D, Wenzel S. *Digitale Fabrik. Methoden und Praxisbeispiele*. Berlin: Springer Vieweg; 2018.
- [5] SDZ GmbH *Tutorial, Part 1 & 2*. Version 8.1. 2020. Online <https://www.sdz.de/downloads/>, 28.02.2020.

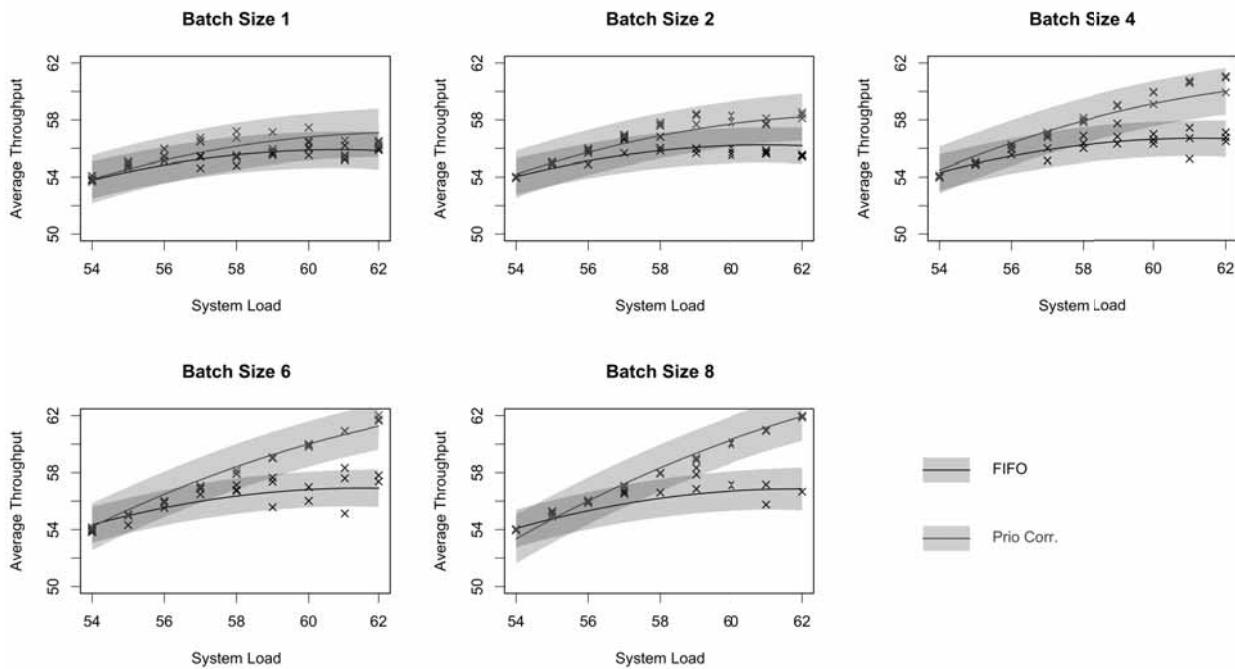


Figure 7: LCCs for *average throughput* with 99% prediction interval and all simulated data points.

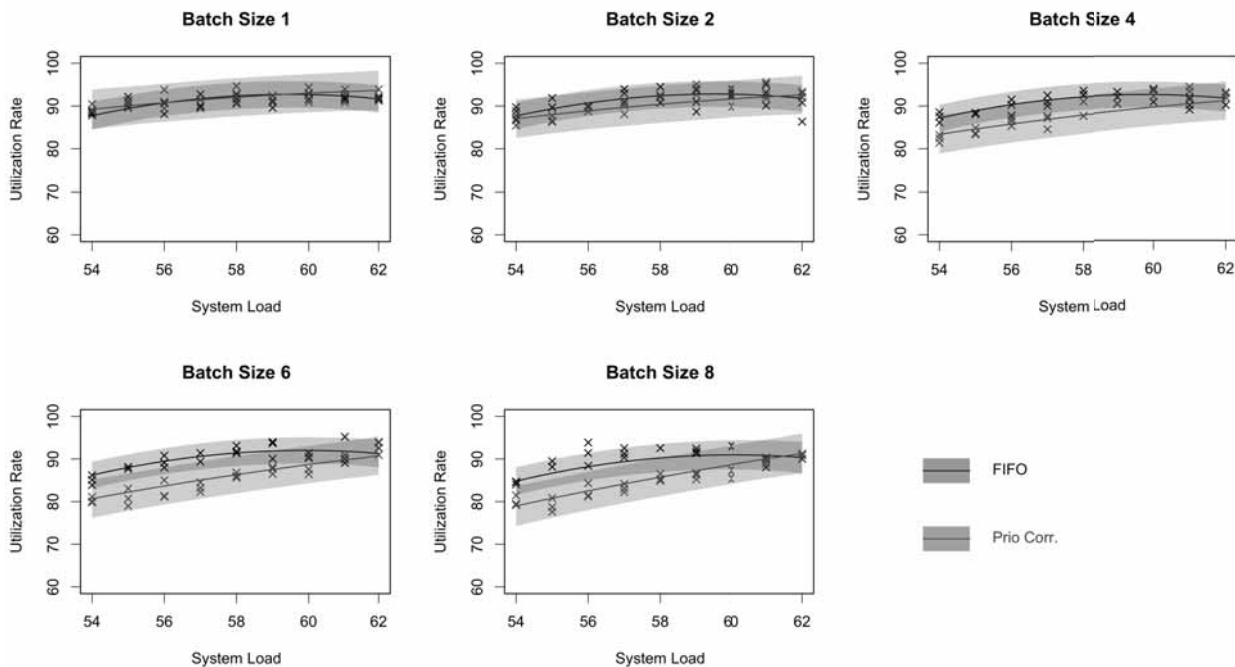


Figure 8: LCCs for *utilization rate* with 99% prediction interval and all simulated data points.

- [6] Nyhuis P, Wiendahl H-P. *Logistische Kennlinien. Grundlagen, Werkzeuge und Anwendungen*. Berlin: Springer; 2003.
- [7] Nyhuis P, Wiendahl H-P. Logistic Production Operating Curves – Basic Model of the Theory of Logistic Operating Curves. *CIRP Annals*. 2006; 55(1): 441–444.
- [8] Nyhuis, P. Practical Applications of Logistic Operating Curves. *CIRP Annals*. 2007; 55(1): 483–486.
- [9] Weigert, G. Operating Curves of Manufacturing Systems: A Theoretical Discourse. In: Azevedo, A, editor. *Advances in Sustainable and Competitive Manufacturing Systems. Lecture Notes in Mechanical Engineering*. Heidelberg: Springer; 2013. 887–897.
- [10] VDI 3633 Blatt 1:2014-12. *Simulation von Logistik-, Materialfluss und Produktionssystemen; Grundlagen (Simulation of systems in materials handling; Logistics and production)*. Berlin: Beuth Verlag. (A list of the currently available parts in this series of standards is available on the Internet at www.vdi.de/3633.)
- [11] Wenzel S, Weiß M, Collisi-Böhmer S, Pitsch H, Rose O. *Qualitätskriterien für die Simulation in Produktion und Logistik*. Berlin Heidelberg: Springer; 2008.
- [12] Wenzel, S. Simulation logistischer Systeme. In: Tempelmeier, H, editor. *Modellierung logistischer Systeme*. Berlin: Springer-Vieweg; 2018. 1–34.
- [13] Santner TJ, Williams BJ, Notz WI. *The Design and Analysis of Computer Experiments*. 2nd ed. New York: Springer; 2018.
- [14] Myers RH, Montgomery DC, Anderson-Cook CM. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. 4th ed. Hoboken, New Jersey: Wiley; 2016.
- [15] Fahrmeir L, Kneib T, Lang S, Marx B. *Regression: Models, Methods and Applications*. Berlin Heidelberg: Springer; 2013.

Metaheuristic Simulation-based Production Planning for Energy Efficiency: A Case Study

Bernhard Heinzl^{*}, Wolfgang Kastner

Research Unit of Automation Systems E191-03, TU Wien, Treitlstraße 1–3, 1040 Vienna, Austria;

^{*}bernhard.heinzl@tuwien.ac.at

SNE 30(3), 2020, 105-116, DOI: 10.11128/sne.30.tn.10523
Received: July 30, 2020; Revised: August 12, 2020;
Accepted: August 15, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Modern industrial production planning and control (PPC) systems are responsible for supporting planning decisions on how to optimally produce a given set of products while minimizing costs and retaining production constraints, such as delivery tardiness or off-times. In recent years, more and more attention has also been paid on energy efficiency as part of production optimization, resulting in competing optimization targets. In order to solve such complex multi-objective scheduling problems in practice, metaheuristic methods are used because of their ability to deliver acceptable solutions in feasible time. In this paper, we demonstrate the application of a General Variable Neighborhood Search (GVNS) metaheuristic on a case study of flow shop scheduling in an industrial bakery in different scenarios and study the effect of different energy prices on the planning result. The case study features a simple production line with thermal processes for baking and freezing and also incorporates the energy supply system as well as a model of the thermal building hull. The metaheuristic is combined with a hybrid discrete/continuous simulation model to evaluate the energy efficiency of different production scenarios. The hybrid simulation enables to accurately capture material and energy flow within the production in an integrated and dynamic manner. Overall, this simulation-based optimization method is intended to support energy-aware production scheduling in practical applications.

Introduction

Energy efficiency in industrial production has become an important topic in recent years because of the substantial potential for energy savings in the industrial sector [1]. Energy-aware Production Planning and Control (PPC) strategies can be used to influence energy demand and energy costs during operation, for example by shifting the production of energy-intensive products to the night hours, where energy is often cheaper. However, it is not sufficient to only consider energy as an optimization goal. Instead, energy efficiency must be seen as part of a multi-objective system of production targets together with production variables such as storage costs, throughput times or delivery delays. Such multi-objective problems with complex, sometimes time-dependent constraints are hard to solve for real-world problems. Modern solutions often rely on heuristic or metaheuristic methods [2].

For evaluating the fitness of solution candidates during metaheuristic search, simulation-based methods are gaining interest because they enable to capture complexity of real-world problems including difficult dynamic interactions without the limiting assumptions many other approaches have. However, with regard to energy optimization, interdisciplinary holistic simulation models are required which include dynamic interactions across engineering domains in order to get an accurate prediction of the overall energy demand, that not only includes production machinery, but also technical building services. For example, heating a production oven generates waste heat that is dissipated into the room and affects heating and cooling energy demand for the building. Similarly, the actual setup time for preheating the oven depends on different conditions, including which products have been produced before, and the setup time affects production throughput and scheduling. Incorporating energy considerations in production logistics simulations with their time-dependent interactions in an accurate manner requires advanced

modeling and simulation approaches that combine discrete (product flow) and continuous dynamics (energy flow). Describing such hybrid discrete/continuous simulation models in a consistent manner with formally sound semantics in complex real-world applications is still challenging [3, 2].

In our work, we aim to support modern PPC tools in incorporating energy considerations into the planning process using simulation-based optimization techniques. We present a metaheuristic procedure for energy-aware optimization of production scheduling, which employs a dynamic discrete/continuous simulation model for evaluating overall energy demand as well as material flow, while taking into account dynamic interdependencies across domains. To this end, we have developed a component-based hybrid simulation tool based on a formal model description, called *hyPDEVs*. The simulation tool includes reusable model components that can be used to simulate interdisciplinary production systems [4]. The optimization method aims at sequencing and time scheduling a given list of production jobs while minimizing energy demand together with other production goals. The procedure combines a Variable Neighborhood Search (VNS) metaheuristic with Variable Neighborhood Descent and Simulated Annealing (SA) for local search and diversification. We demonstrate the feasibility of this method on a flow shop scheduling problem of an industrial bakery. The simulation model includes a production line with alternative paths as well as technical building services for energy supply and a thermal building model. We compare different scenarios, investigate the effect of variable energy prices and on the planning result and highlight the potential benefit of considering energy as an optimization target.

1 Related Work

Since optimization in a complex solution space with simultaneous objectives is a difficult problem to solve, practical multi-objective optimization has spawned a wealth of approaches and solution methods [5, 6]. A general overview is given in [7, 8, 9], while [10] focuses on optimization in the food manufacturing industry. Optimization methods can be categorized in exact methods, such as gradient methods, dynamic programming, etc., and approximate/heuristic methods like greedy heuristics, Simulated Annealing, Evolutionary Algorithms or Variable Neighborhood Search [11].

According to [10], metaheuristics and customized multi-objective heuristic approaches are well-suited for applications in real-life industrial production planning problems (which typically are NP-hard), in contrast to exact approaches that require simplified models. Metaheuristics allow to explore the search space more efficiently and effectively, especially if they are tailored to the individual problem [12]. Different metaheuristic algorithms, such as Evolutionary Computation, Tabu Search, Particle Swarm Optimization (PSO) or Simulated Annealing (SA) have been successfully applied to various logistics optimization problems [13]. In previous work, we have also investigated applying a Genetic Algorithm on a similar case study [14, 15]. A set of tuning and customization measures was applied to significantly improve performance, including adapting operators for a guided search, and hybridization with Tabu Search and Pattern Search.

Besides population-based metaheuristics, like Genetic Algorithms, which work with a population of candidate solutions to concurrently sample different regions of the solution space, single-solution-based methods, also called trajectory methods, iterate over a single solution are more exploitation-oriented and usually need fewer simulation evaluations, which improves overall computation time. Among these trajectory methods, Variable Neighborhood Search (VNS) algorithms have shown excellent capability for solving scheduling problems [16]. This is in accordance with other publications, e.g. [17, 18, 16], which have successfully applied VNS for job scheduling problems in the production domain. In [19], the authors compare different optimization methods for simulation-based optimization of production plans, in which VNS also leads to the best results.

For simulation-based optimization strategies in particular, several authors employ discrete-event simulations (DES). In [20], the authors combine a DES with a Genetic Algorithm for energy-oriented machine allocation planning. The DES, however, uses deterministic energy profiles and only allows very simplified consideration of dynamic energetic interdependencies. It also does not include production periphery or building facilities in the energy consumption. Other publications describe hybrid discrete/continuous simulation for energy efficiency in production, however, most of these use them in a scenario-based manner without systematic metaheuristic optimization. In [21], a multilevel simulation is presented to model the material flow to-

gether with production equipment and energy system. This includes dynamic coupling across domains, however, optimizations are conducted manually.

2 Case Study Overview

To evaluate the described method on a real-life example, we devised a case study of a production plant. The case study is a simplified model of a real production plant of an industrial bakery that produces baked goods [4]. The model features a typical production line with machines, storage and conveyor belts, an energy supply system with heater and cooler, and a building model with thermal zones.

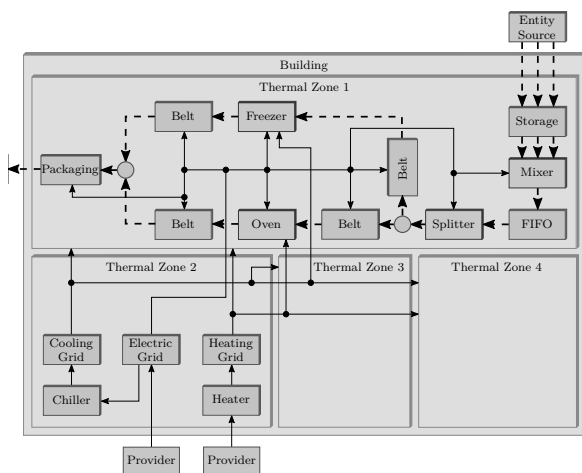


Figure 1: Production facility consisting of production machines (blue), logistics components (purple), technical building services for energy supply (red) and thermal building zones (green).

The model has been simplified compared to the original plant by omitting unnecessary production steps and other components, in order to reduce problem complexity and simplify parametrization. However, this does not change the structure or topology of the general archetype and therefore does not undermine the goal of demonstrating the feasibility of the method. The main critical elements of the production still remain in the model, specifically: intertwined continuous and discrete dynamics, cross-domain modeling with dynamic interdependencies, and complex product flow with splitting, merging and batching. More details on the simulation are given in Section 4.

The production and logistics components form a production line for two product variants: baked and frozen. Baked products pass a oven for baking while frozen products are frozen directly (in a freezer) without being baked. Both are designed as conveyor belts, meaning that new entities continuously enter the stations and leave on the other side. Since these products share all other stations, only one type of product can be produced at any time. For both products, respective ingredients are being pulled from the storage, after which they are mixed into a dough, divided into portions (splitting) and continue on different conveyor belts. After baking/freezing, the finished products are packaged in different quantities.

The building is modelled as a simple thermal compartment model with four thermal zones, each representing a distinct part of the facility: production hall, plant rooms and cold storage. These zones all have independent conditioning (for example, the cold storage is kept at 4 °C) and exchange thermal energy with one another according to the defined wall topology. They also exchange thermal energy with the environment, for which a variable ambient temperature may be specified.

The energy system provides necessary technical building services, mainly supplying energy for the production machines as well as for heating and cooling the thermal building zones. The energy system is comprised of a heater (powered by natural gas) that supplies heat to the oven and building, a chiller (powered by electricity) that supplies cold to the freezer and building, as well as respective energy grids responsible for distributing the energy. The heating and cooling grids also include thermal energy storage that models the storage capacity of the grid. All other production stations receive electric energy from the grid.

The production orders are executed according to a production schedule (Pplan), which is the main input vector to specify the production scenario and is the subject of optimization, which constitutes a flow shop scheduling problem. In addition to the order starting times, the Pplan also specifies the start of the setup processes for the oven and freezer. During these setup processes, the oven (or freezer) is preheated (or cooled) to the defined operating temperature before the products arrive at the station. This also implies that, in contrast to traditional discrete-event material flow simulations, the setup time is not a fixed parameter, but may change dynamically during the optimization process.

The case study further includes a work plan (Aplan) that specified the production steps and process parameters, such as temperature set points, baking time or batching size. It serves as a look-up table for the individual stations and the process parameters may be different for different product types.

3 Optimization Method

Production optimization deals with finding an optimal combination of production resources, such as equipment, utilities or energy, to achieve a given production target in the best possible way. Operative Production Planning and Control (PPC) methods consider production resources and cost factors to find optimal production schedules. To support PPC in practice, modern APS (Advanced Planning and Scheduling) software systems offer integrated resource planning [22]. The complexity of the underlying optimization problem with multiple competing objectives and complex constraint conditions suggests using a simulation-based strategy [23]. Hereby, a simulation model of the system under consideration is part of the objective function and serves as a prescriptive tool to predict and evaluate the performance of a given scenario. Dynamic simulation allows to consider more complex systems than with conventional analytical models, especially for highly time-dependent problems, while offering more accurate predictions and overall improving planning quality [2]. Recent advances in computational techniques have also led to an increased interest in simulation-based methodologies to solve optimization problems [23].

However, the fact that these methods in general do not provide closed analytical representations of the objective function (or its derivatives) prevents straightforward deployment of many standard optimization algorithms. Instead, many practical simulation-based optimization solutions employ metaheuristics that rely solely on the evaluation of the objective function itself. These algorithms modify a candidate solution in an iterative manner to find a near-optimal solution until termination criteria are met. The final solution may not be the global optimum, but is often good enough in practice.

The overall computation cycle of using the simulation-based optimization methodology for production scheduling in industry is illustrated in Figure 2. The starting point is a given demand plan (Dplan) specifying how many of which entities (i.e. products) need to be delivered when. This Dplan serves as a basis

to generate an initial solution of a production schedule (Pplan), which is then evaluated using a dynamic simulation model. The results of the simulation are fed back to the optimization to be evaluated for its fitness using a specified cost function $f(x)$ (objective function). Based on this evaluation, the optimization algorithm iteratively adapts the solution in order to find the minimum of $f(x)$ subject to constraints. The cycle continues until certain termination criteria (e.g. fixed number of iterations, computation time threshold, etc.) are met [2].

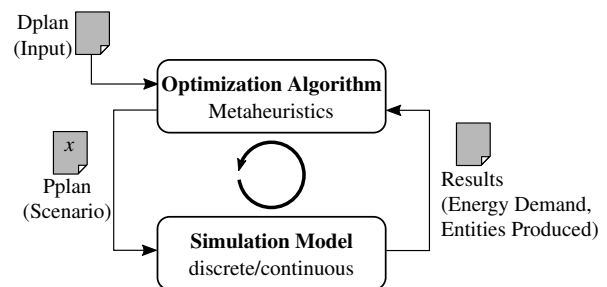


Figure 2: Simulation optimization cycle. Input is a demand plan (Dplan) of entities to be produced. The optimization algorithm generates a production schedule (Pplan), which is adapted iteratively based on an evaluation of simulation results [2].

We employ a single-solution metaheuristic method based on a General Variable Neighborhood Search (VNS) procedure. The VNS [16] is an effective method guiding a local search by switching between increasingly larger neighborhoods to efficiently explore the solution space. The VNS consists of (1) a shaking phase that generates randomized perturbations of the solution for diversifying the search to escape local optima, and (2) an intensification phase that searches for improvements in the local neighborhood, typically using a local search procedure. For improving the performance of the search, the Generalized VNS (GVNS) replaces the local search by a Variable Neighborhood Descent (VND) procedure [24, 25]. The VND is similar to the VNS, but does not include shaking and thus limits the search to strict improvements (i.e. descent). It does not use the same neighborhood structures as the VNS, which offers more flexibility in tailoring the search procedure to the problem instance [2].

Algorithm 1 presents an overview of the optimization method as pseudocode. After generating a feasible initial solution x , the VNS component (described in more detail in Section 3.3) guides a search for im-

provements over a fixed number of iterations n_{iter} by generating in each iteration a random neighboring solution x' (by means of the neighborhood structure $\mathcal{N}_k(x)$) that serves as starting point for the VND that tries to find a local minimum x'' . If the solution is accepted, the VNS restarts with neighborhood \mathcal{N}_1 and the new starting solution x'' . Otherwise, the neighborhood is enlarged. The acceptance criterion is based on Simulated Annealing (SA) [24] for further diversifying the search by allowing to accept potentially worse solutions during the early search phase. Infeasible solutions are also allowed during the search and are evaluated based on a penalizing cost function, which is described in Section 3.2. For the particular case study, the optimization vector x contains the job starting times and the setup time duration for oven and freezer, which allows to optimize job sequencing and scheduling as well as setup processes, which also have an influence on the energy demand.

Algorithm 1 GVNS/SA algorithm

```

1:  $\mathcal{N}_k \leftarrow$  set of VNS neighborhood structures
2:  $x \leftarrow \text{InitialSolution}()$ 
3:  $k \leftarrow 1$ 
4:  $i \leftarrow 0$ 
5:  $T \leftarrow T_0$ 
6: while  $i \leq n_{\text{iter}}$  do
7:    $x' \leftarrow \text{shaking}(\mathcal{N}_k(x))$   $\triangleright$  random perturbation
8:    $x'' \leftarrow \text{VND}(x', \mathcal{N}_k)$   $\triangleright$  local improvement search
9:   if  $\text{acceptSA}(x'', x, T)$  then  $\triangleright$  SA criterion
10:     $x \leftarrow x''$ 
11:     $k \leftarrow 1$ 
12:   else
13:     $k \leftarrow (k \bmod k_{\text{max}}) + 1$   $\triangleright$  next neighborhood
14:   end if
15:    $i \leftarrow i + 1$ 
16:   update  $T$ 
17: end while
  
```

3.1 Initial Solution

An initial solution (function *InitialSolution()*) is generated by employing a custom construction heuristic, which takes the jobs from the provided input list (Dplan), sorts them in increasing order of due date and then schedules the jobs in this order as soon as possible (forward scheduling) while considering safety gaps between jobs as well as setup times. This method en-

sures that no collisions occur, and the initial solution is feasible to be a valid starting point for the subsequent improvement search. However, further constraint violations, such as delivery delay, or energy considerations, are not checked during this phase.

3.2 Generalized Cost Function

The algorithm allows infeasible solution during the search process. Constraints are penalized by means of the generalized cost function

$$f(x) = \omega_1 \cdot f_{\text{dev}}(x) + \omega_2 \cdot f_{\text{en}}(x) + \omega_3 \cdot f_{\text{st}}(x) + \omega_4 \cdot f_{\text{del}}(x) + \omega_5 \cdot f_{\text{sep}}(x) \quad (1)$$

where f_{dev} denotes a penalization for deviating number of entities produced, f_{en} the energy costs, f_{st} the storage costs, f_{del} is a penalization for any potential delivery delay, and f_{sep} penalizes job separation violations. These part goals are evaluated based on the simulation.

The energy costs f_{en} are calculated by taking the simulated power supply P_s from each of the external providers $s \in \text{Prov}$, rate them with time-dependent energy prices c_s and accumulate them over time to obtain the overall energy costs:

$$f_{\text{en}}(x) = \sum_{s \in \text{Prov}} \sum_{t_i} P_{s,t_i}(x) \cdot (t_{i+1} - t_i) \cdot c_s(t_i). \quad (2)$$

This method of using variable energy prices allows to take into account different effects that have an influence on the energy price and therefore may potentially influence the production planning result, like for example lower energy prices during nighttime, or an additional photovoltaic system that provides solar energy during the day. The storage costs f_{st} are determined based on the time difference between job completion and delivery due date (given in the demand plan). For more details, we also refer to [26]. The coefficients $\omega_i > 0$ are weighting factors, which may be adapted by the user to balance their preferences for individual part goals and trade-offs in a transparent manner. This weighted sum method is common for multi-objective optimization problems in practice as it is easy to implement and intuitive for the user. However, the results are often highly dependent on the weights [9, 2].

3.3 Neighborhood Structures

The general operation of the VNS has been described in Algorithm 1. It is responsible for diversifying the

search in a structured way during the shaking phase by successively changing the neighborhood structures \mathcal{N}_k . These neighborhood structures are usually defined implicitly by means of operators that modify the solution. For our case, we have defined four different operators [2]:

1. *OpSwitch*: This operator changes the order of jobs by taking a random number r of successive jobs (starting from a random position) and moving them to a different position. Hereby, r is chosen in the interval $r \in [1, \min\{r_{\max}, n\}]$, where n is the overall number of jobs and r_{\max} changes depending on the neighborhood k , see Table 1.
2. *OpShift*: The shifting operator takes a random position and moves all subsequent jobs by a specified time t_{shift} , where t_{shift} depends on k .
3. *OpChangeSetuptime*: Here, the setup time is changed by a specified amount t_{setup} .
4. *OpMerge*: The merging operator takes two random jobs, which are removed by a distance of d , and which have the same product type and combines them into one job. Merging jobs has the advantage that it reduces the number of setup processes and avoids gaps between jobs, thereby increasing production and energy efficiency.

The VND, which is used in place of a local search procedure inside the VNS to improve the generated solution in the intensification phase [25], uses the same basic operations, albeit in different neighborhoods depending on the current VNS neighborhood $\mathcal{N}_k(x)$. For the switching operator, the VND neighborhood containing all pairs of successive jobs is explored exhaustively. The same is done with respect to merging. For *OpShift*, the VND checks shifting groups of jobs in a binary search pattern by a set of different times t_{shift} . Similarly for the setup times, which are being reduced iteratively by a set of different t_{setup} . The complete neighborhood structures used for the case study are presented in Table 1.

3.4 Acceptance Criterion

Instead of accepting only improving solutions, a modified acceptance criterion (*acceptSA()*) is used that is based on a Simulated Annealing (SA) method [24, 2]. It enables to further diversify the search and better escape local optima by accepting potentially worse solutions, albeit with decreasing probability as the search

k	Operator	Shaking	VND
1	1	$r_{\max} = 2$	$r_{\max} = 1$
2	1	$r_{\max} = 4$	$r_{\max} = 1$
3	2	$t_{\text{shift}} = 8 \text{ h}$	$t_{\text{shift}} \in \{4, 2, -1, 0.5\} \text{ h}$
4	2	$t_{\text{shift}} = 12 \text{ h}$	$t_{\text{shift}} \in \{4, 2, -1, 0.5\} \text{ h}$
5	3	$t_{\text{setup}} = 0.5 \text{ h}$	$t_{\text{setup}} \in \{0.5, 0.25\} \text{ h}$
6	4	$d = 2$	$d = 1$

Table 1: Neighborhood structures used in the GVNS.

progresses. To be more precise, while an improving solution is always accepted, a deteriorating solution x'' is accepted with the probability

$$p_{\text{SA}} = \exp(-(f(x'') - f(x))/T), \quad (3)$$

where f is the generalized cost function (see Section 3.2) and T is the temperature that decreases linearly after every VNS iteration in such a way that $T < 10^{-3}$ during the last 10% of iterations. This effectively tightens the acceptance criterion as the search progresses until, finally, only improving solutions are accepted at the end. The temperature T is initialized with T_0 according to $T_0 = -\Delta_{\text{SA}}/\log(0.5)$, meaning that, initially, a solution being Δ_{SA} worse than $f(x)$ is accepted with a probability of 50%.

4 Hybrid Simulation

Hybrid discrete/continuous simulation in the context of interdisciplinary assessment of energy efficiency in production enables both the material flow to be accurately modelled as Discrete-Event system and the energy flow by means of differential equations, while also taking into account dynamic interactions between these domains. Continuous representation of energy flow, as opposed to discrete energy profiles, enables to accurately incorporate transient dynamics, for example the heat-up process of an oven or the thermal heat capacity of the building. For hybrid modeling and simulation, we employ a formal model description, called hypPDEVS, which is based on the Discrete Event System Specification (DEVS) [27].

The hypPDEVS formalism extension aims to incorporate continuous model aspects into a discrete-event model description [28]. It is similar to the DEV&DESS formalism introduced by Prähofer [27], with the difference being that it allows improved handling of parallel and concurrent events (since it is based on the Parallel-DEVS extension [27]).

DEVS formalisms follow a component-based paradigm where the overall model is comprised of individual components being coupled together. Coupled systems can be arranged hierarchically, meaning they can be incorporated just like an atomic into a larger coupled system. This property allows to construct modular hierarchical (tree-like) models in a component-based manner [29, 27].

Component-based modeling as a paradigm facilitates modularity and separation of concerns for managing the complexity of large-scale simulation models. Components can be implemented as object classes in an object-oriented programming language [30]. These classes can then be organized in a library of model components, to be instantiated by the user in different contexts, thereby facilitating component reuse. Reusing models is crucial in an attempt to reduce the effort necessary to develop new application models [31, 32]. In order to retain modularity and reusability, it is strictly necessary to encapsulate all aspects (in particular discrete and continuous aspects) of a model component within uniform boundaries. This may become a challenge when using e.g. co-simulation [33], as this approach forces the user to split the model into discrete and continuous sub-models to be implemented by respective simulators [34]. The hypPDEVs specification, on the other hand, allows to integrate hybrid aspects already on the atomic level, thereby removing the restrictions on hybrid composability.

Based on the described hypPDEVs formalism, a simulation tool was implemented in C++ [4] together with a library of model components for modeling production systems, which are also used in the case study, see Figure 1. The production stations, especially oven and freezer, are modelled as hybrid discrete/continuous models, where the discrete model is responsible for entity flow and control logic, and the continuous model handles energy input and conversion. In particular, the energy conversion follows simple energy balance equations, including thermal heat capacity of the station, and generates diffuse waste heat that is dissipated into the respective thermal zone in the room model. Energy demand for oven and freezer is controlled by a PI controller located inside the component.

The energy model also interacts with the material flow by means of state events that indicate e.g. when the oven has reached its target temperature and is ready to accept entities. More details are described in [29].

The building and energy system components on the

other hand are mainly continuous models (as they do not directly interact with entities), except for simple control logics. The building model contains four thermal zones with homogeneous temperature distribution and calculates heat transfer across the walls between the zones as well as with the environment. For this, a temperature profile for the ambient temperature can be specified as simulation input, which allows to compare different weather conditions, e.g. summer vs. winter. For this case study, the ambient temperature is set to 20 °C. A more sophisticated thermal building model has also been developed for the hypPDEVs simulator [35], which is, however, not part of this case study. Outside energy providers supply the energy across the system boundary to the energy system, which is billed to the customer using time-dependent energy pricing.

5 Case Study Experiments

To demonstrate the application of the proposed GVN-S/SA method on the flow shop scheduling case study, the optimization was implemented as a prototype in MATLAB and coupled with the standalone hypPDEVs simulator [36]. Table 2 presents a simplified scenario of demand (Dplan) needing to be scheduled over the course of two days (i.e. 48 h simulation time). The table lists different orders (coming from customers) with quantities and delivery due dates (measured from the start of the simulation). Although realistic demand plans are usually more complex, simplifying the scenario allows to manually verify the results and check the algorithm for plausibility.

For the ambient temperature, we consider a real-world temperature curve presented in Figure 3 and we compare different energy price profiles $c_s(t)$ (see Equation (2)). The price profiles are depicted in Figure 4. One electricity price profile $c_{1,real}(t)$ was taken from real historical electricity spot market data of the Austrian Energy Exchange (EXAA)¹, the other one $c_{1,const}(t)$ represents its mean value (0.0437 €/kWh) kept constant over the entire period. The gas price $c_2(t)$ is also constant with 0.04 €/kWh.

Figure 5 depicts the Oven and Freezer allocations (i.e. number of entities and temperature over time) of the final optimization result when using the constant energy price profile. The optimization ran for $n_{iter} = 50$ iterations and the partgoal weights were cho-

¹Source: <https://www.exaa.at/en/marketdata/historical-data>

Type	Quantity	Due Time	
		Scenario 1	Scenario 2
baked	32	48 h	48 h
baked	8	24 h	—
frozen	20	48 h	—
baked	4	48 h	—

Table 2: Demand plan for the simulated scenario.

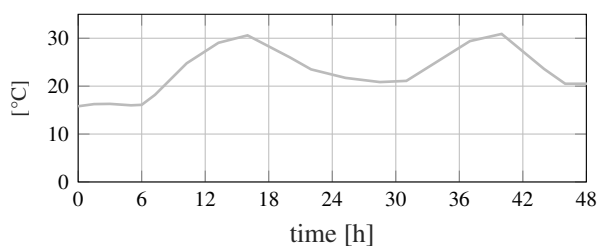


Figure 3: Ambient temperature curve used for the simulation, based on real data.

sen as $\omega = (\omega_i)_{i=1..5} = (1, 2, 2, 1, 1)$, while the deterioration percentage of the Simulated Annealing (SA) acceptance criterion was lowered to $\Delta_{SA} = 0.3$.

The result is by and large as expected, with all jobs being produced as late as possible (due to the storage costs) and the one job (Job 2 in Table 2) adhering to the earlier delivery due date. While the overall goal was improved by 28%, not much energy was saved in this case, only about 6%, as there is not much leeway for shifting or merging jobs. This is also visible in the cost function plot in Figure 6. While the energy costs make up the majority of the target value, most of the improvements are achieved through the storage costs, which have been lowered by 77%.

Also, the variable ambient temperature itself does

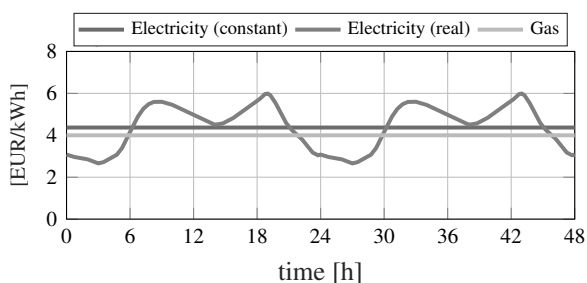


Figure 4: Energy price profiles, constant and real.

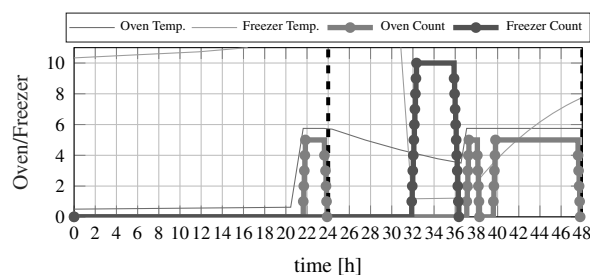


Figure 5: Optimization results for Scenario 1. The dashed vertical lines indicate job due times.

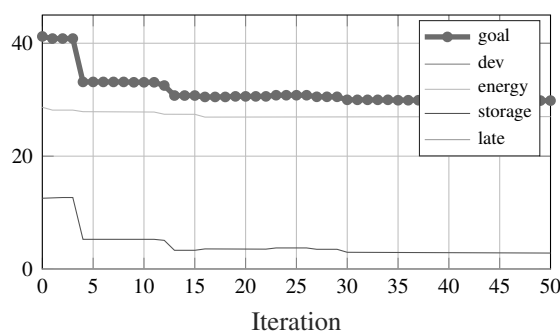


Figure 6: Cost function and part goals for Scenario 1.

not have much influence on the optimization result either. As further analyses have shown, this is partly due to the temperature control in the building. On the one hand, this suppresses any direct influence of the ambient temperature on the thermal behavior of the production stations, on the other hand, a (more or less) constant total amount of energy has to be supplied to heat the building over the observation period, which is only reduced by the waste heat of the stations. However, it is irrelevant when exactly this waste heat is supplied.

Next, we want to take a deeper look at the influence the energy has on the planning result. For this, we omit the storage costs in the target system by setting the respective weight to $\omega_3 = 0$. This has the effect that the oven lots can be produced earlier, together with Job 2 that is due earlier, thereby saving a second setup process. Switching the sequence of the oven lots does not make much of a difference energetically. The result, which we will call x_c for further reference, is shown in Figure 7.

If we compare this result with the planning result x_r obtained with the real energy price profile $c_{1,real}(t)$, shown in Figure 8, a different picture emerges. Here, the jobs are scheduled earlier, especially in the periods

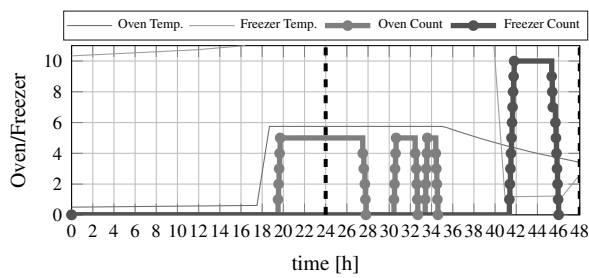


Figure 7: Oven and Freezer allocation x_c for Scenario 1 without storage costs and constant energy price.

with low energy prices, while the periods of high energy prices are avoided. Especially the oven lots exploit these valleys, even though the oven continues to run during these gaps.

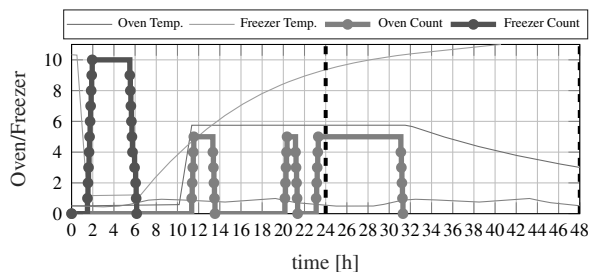


Figure 8: Results x_r for Scenario 1 without storage costs and real energy price (red line).

If we take x_c (Figure 7), which only considers a constant energy price, and evaluate it with the realistic variable price profile $c_{1,\text{real}}(t)$ (as it would be done in reality), we see that the goal would be about 5% worse compared to x_r , which considers the real energy price in the optimization, see Table 3.

	var. Price x_r	const. Price x_c	Difference
$f(x)$	27.45	28.82	5%

Table 3: Comparison of Scenario 1 between x_r and x_c , showing the cost value evaluated using the real energy price profile $c_{1,\text{real}}(t)$.

The progression of the cost function, depicted in Figure 9, shows that, compared to the results without storage costs, it takes more iterations to reach a near-optimal value, and the overall values are slightly higher. This will become even more apparent later in Figure 10.

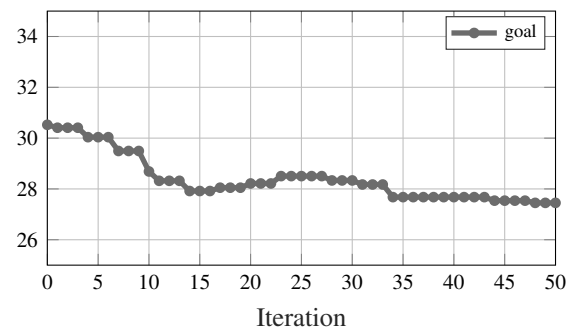


Figure 9: Cost function, i.e. energy costs, for Scenario 1 without storage costs and real energy price.

The effect of job scheduling depending on the energy price can be examined more closely by looking at a single job as defined in Scenario 2 in Table 2. By sweeping a single job across the planning horizon (starting time $x_1 = 2 \text{ h} \dots 37 \text{ h}$), we can determine how the target function changes depending on the starting time. The result is presented in Figure 10. Both with constant energy price and real price profile, the target function has clearly pronounced hills and valleys that correlate approximately with the ambient temperature. However, the optimum values do not align, namely $x_1 = 27 \text{ h}$ compared to $x_1 = 22 \text{ h}$ with constant energy price.

What is also noticeable is that the target value is on average about 9 % higher when using the real energy profile, compared to the constant value, even though the mean price over the course of the day is the same, see Figure 4. These higher energy costs have to do with the fact that the energy demand is unevenly distributed throughout the day, not only due to production, but also, for example, due to intermittent filling of the heat and cold storage (that are part of the energy system infrastructure), which often occurs in times of above-average energy prices. Even if one could have suspected that this would balance out over the day, this does not seem to be the case here [36].

6 Conclusion

The case study results show that including energy considerations and realistic energy pricing into the planning optimization can potentially reduce overall costs and provide a better planning result. Although the case study has been simplified to highlight the essential characteristics, it is easy to imagine applying the described method to more complex production systems with more

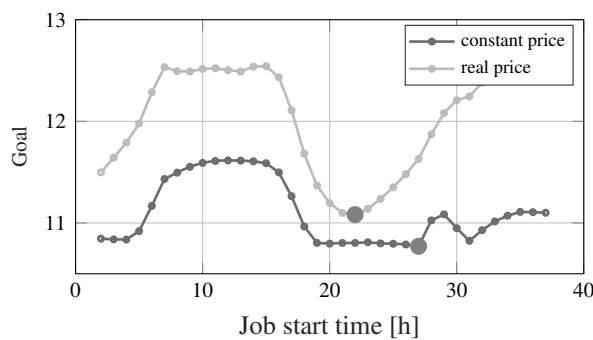


Figure 10: Target value plotted over the start time sweep, both for constant energy price and real energy price profile. The red dots indicate the respective minima.

diverse material flow and product types, as part of a practical PPC tool. This makes the optimization problem even more interesting because of the job completion times being more difficult to predict due to potential bottlenecks etc. From the current point of view, the presented simulation-based approach should also be applicable for these cases.

Many factors potentially influence the overall energy demand, which have to be included into the simulation. The building facilities contribute a significant portion to that, but also setup times of thermal processes can be optimized in order to not only save energy, but also improve production tardiness and reduce storage costs. These factors, however, have to be considered simultaneously, as part of a multi-objective optimization, instead of individually.

One advantage of the VNS metaheuristic is that it allows to tailor the operators and neighborhood structures to the individual problem instance, thereby allowing more efficient search and decreasing computation time. Neighborhoods (i.e. switching, shifting, etc.) are explored repeatedly and iteratively instead of sequentially, and the trade-off between exploration and exploitation can be controlled by the user. In contrast to population-based approaches, like Genetic Algorithms, VNS as a single-solution-based methods needs fewer function evaluations (i.e. simulation runs), which is a topic of importance for simulation-based methods. In the future, more extensive comparisons to population-based methods are planned.

Experiments on the case study have also shown that the Simulated Annealing (SA) acceptance criterion is

important especially for the merging operator, since, although merging two jobs might be beneficial in the end, there might still be a short-term increase in the cost function (e.g. if the products from the second job are being produced sooner and therefore increase storage costs). In the future, VNS parameter and neighborhood calibration might still be improved and even be automated to directly adjust to the problem instance without the need for user intervention.

The hyPDEVS-based simulation itself, despite its hybrid nature, delivers sufficient performance to be feasible for simulation-based optimization tasks with a large number of iterations. This has also been tested with larger real-world case studies, see e.g. [37].

Acknowledgement

This work is funded by the Austrian Research Promotion Agency (FFG) as part of the project AS-PeCT (Adaptive Smoothed Production, project number: 858655) within the research program "Production of the Future". The authors would like to thank all project partners for their contributions.

References

- [1] Chan Y, Kantamaneni R, Allington M. Study on Energy Efficiency and Energy Saving Potential in Industry and on Possible Policy Mechanisms. *Tech. rep.*, ICF Cons. Ltd. 2015.
- [2] Heinzl B, Kastner W. A General Variable Neighborhood Search for Simulation-Based Energy-Aware Flow Shop Scheduling. In: *Proceedings of the 2020 Summer Simulation Conference*. Virtual Event: SCS. 2020; .
- [3] 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.
- [4] Heinzl B, Kastner W. Platform-Independent Modeling for Simulation-Based Energy Optimization in Industrial Production. *International Journal of Simulation: Systems, Science and Technology*. 2019; 20(6):10.1–10.10.
- [5] Hwang CL, Masud ASM. *Multiple Objective Decision Making — Methods and Applications: A State-of-the-Art Survey*. Springer Science & Business Media. 2012.
- [6] Ehrgott M, Gandibleux X. *Multiobjective Combinatorial Optimization — Theory, Methodology,*

- and Applications. In: *Multiple Criteria Optimization: State of the Art Annotated Bibliographic Surveys*, edited by Ehrgott M, Gandibleux X, International Series in Operations Research & Management Science, pp. 369–444. Boston, MA: Springer US. 2002;.
- [7] Boussaïd I, Lepagnot J, Siarry P. A Survey on Optimization Metaheuristics. *Information Sciences*. 2013;237:82–117.
- [8] Xiujuan L, Zhongke S. Overview of Multi-Objective Optimization Methods. *Journal of Systems Engineering and Electronics*. 2004;15(2):142–146.
- [9] Freitas AA. A Critical Review of Multi-Objective Optimization in Data Mining: A Position Paper. *ACM SIGKDD Explorations Newsletter*. 2004;6(2):77–86.
- [10] Wari E, Zhu W. A Survey on Metaheuristics for Optimization in Food Manufacturing Industry. *Applied Soft Computing*. 2016;46:328–343.
- [11] Karimi-Nasab M, Aryanezhad M. A Multi-Objective Production Smoothing Model with Compressible Operating Times. *Applied Mathematical Modelling*. 2011;35(7):3596–3610.
- [12] Güller M, Uygun Y, Noche B. Simulation-Based Optimization for a Capacitated Multi-Echelon Production-Inventory System. *Journal of Simulation*. 2015;9(4):325–336.
- [13] Silva CA, Runkler TA, Sousa JM, da Costa JMS. Optimization of Logistic Processes in Supply-Chains Using Meta-Heuristics. In: *Progress in Artificial Intelligence*, edited by Goos G, Hartmanis J, van Leeuwen J, Pires FM, Abreu S, vol. 2902 of *Lecture Notes in Computer Science (LNCS, Volume 2902)*, pp. 9–23. Berlin, Heidelberg: Springer Berlin Heidelberg. 2003;.
- [14] Sihn W, Sobottka T, Heinzl B, Kamhuber F. Interdisciplinary Multi-Criteria Optimization Using Hybrid Simulation to Pursue Energy Efficiency through Production Planning. *CIRP Annals – Manufacturing Technology*. 2018;67(1):447–450.
- [15] Sobottka T, Kamhuber F, Sihn W. Increasing Energy Efficiency in Production Environments through an Optimized, Hybrid Simulation-Based Planning of Production and Its Periphery. In: *Proceedings of the 24th CIRP Conference on Life Cycle Engineering*. Kamakura, Japan. 2017; pp. 440–445.
- [16] Roshanaei V, Naderi B, Jolai F, Khalili M. A Variable Neighborhood Search for Job Shop Scheduling with Set-up Times to Minimize Makespan. *Future Generation Computer Systems*. 2009;25(6):654–661.
- [17] Yazdani M, Amiri M, Zandieh M. Flexible Job-Shop Scheduling with Parallel Variable Neighborhood Search Algorithm. *Expert Systems with Applications*. 2010; 37(1):678–687.
- [18] Adibi MA, Zandieh M, Amiri M. Multi-Objective Scheduling of Dynamic Job Shop Using Variable Neighborhood Search. *Expert Systems with Applications*. 2010;37(1):282–287.
- [19] Gansterer M, Almeder C, Hartl RF. Simulation-Based Optimization Methods for Setting Production Planning Parameters. *International Journal of Production Economics*. 2014;151.
- [20] Rager M. *Energieorientierte Produktionsplanung: Analyse, Konzeption und Umsetzung*. Springer-Verlag. 2008.
- [21] Thiede S, Schönemann M, Kurle D, Herrmann C. Multi-Level Simulation in Manufacturing Companies: The Water-Energy Nexus Case. *Journal of Cleaner Production*. 2016;139:1118–1127.
- [22] Kilger C, Meyr H, Stadler H. *Supply Chain Management and Advanced Planning: Concepts, Models, Software, and Case Studies*. Springer. 2015.
- [23] Swisher J, Hyden P, Jacobson S, Schruben L. A Survey of Simulation Optimization Techniques and Procedures. In: *Proceedings of the 2000 Winter Simulation Conference*, vol. 1. Orlando, FL, USA: IEEE. 2000; pp. 119–128.
- [24] Siarry P. *Metaheuristics*. Springer. 2016.
- [25] Hansen P, Mladenović N. Variable Neighborhood Search Methods. In: *Encyclopedia of Optimization*, edited by Floudas CA, Pardalos PM, pp. 3975–3989. Boston, MA: Springer US. 2009;.
- [26] Kamhuber F, Sobottka T, Heinzl B, Sihn W. An Efficient Multi-Objective Hybrid Simheuristic Approach for Advanced Rolling Horizon Production Planning. In: *Proceedings of the 2019 Winter Simulation Conference*, vol. 1. Maryland, USA: IEEE. 2019; pp. 1–11.
- [27] Zeigler BP, Prähofner H, Kim TG. *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*. USA: Academic Press. 2000.
- [28] Deatcu C, Pawletta T. A Qualitative Comparison of Two Hybrid DEVS Approaches. *SNE - Simulation Notes Europe*. 2012;22(1):15–24.
- [29] Heinzl B. Hybrid Modeling of Production Systems: Co-Simulation and DEVS-Based Approach. Diploma Thesis, TU Wien, Vienna, Austria. 2016.
- [30] Leobner I. Modeling of Energy Systems for Complex Simulations. Dissertation, TU Wien, Wien. 2016.

- [31] Crnkovic I, Chaudron M, Larsson S. Component-Based Development Process and Component Lifecycle. In: *International Conference on Software Engineering Advances*. 2006; .
- [32] Cetinkaya D, Verbraeck A, Seck MD. Applying a Model Driven Approach to Component Based Modeling and Simulation. In: *Proceedings of the Winter Simulation Conference, WSC '10*. Baltimore, Maryland. 2010; pp. 546–553.
- [33] Awais MU. Distributed Hybrid Co-Simulation. Dissertation, TU Wien, Wien. 2015.
- [34] Heinzl B, Raich P, Preysner F, Kastner W. Simulation-Based Assessment of Energy Efficiency in Industry: Comparison of Hybrid Simulation Approaches. In: *Proceedings of the 9th Vienna International Conference on Mathematical Modelling (MATHMOD 2018)*. Wien, Austria. 2018; .
- [35] Smolek P, Leobner I, Gourelis G, Mörzinger B, Heinzl B, Ponweiser K. Hybrid Building Performance Simulation Models for Industrial Energy Efficiency Applications. *Journal of Sustainable Development of Energy, Water and Environment Systems*. 2018;.
- [36] Heinzl B. Methods for Hybrid Modeling and Simulation-Based Optimization in Energy-Aware Production Planning. PhD Thesis, TU Wien, Wien. 2020.
- [37] Sobottka T, Kamhuber F, Faezirad M, Sihn W. Potential for Machine Learning in Optimized Production Planning with Hybrid Simulation. *Procedia Manufacturing*. 2019;39:1844–1853.

Reduction of Complexity in Q-Learning a Robot Control for an Assembly Cell by using Multiple Agents

Georg Kunert^{1*}, Thorsten Pawletta¹, Sven Hartmann²

¹Research Group Computational Engineering and Automation, Wismar University of Applied Sciences: Technology Business and Design, Philipp-Müller-Straße 14, D-23966 Wismar, Germany; *georg.kunert@cea-wismar.de

²Department of Informatics, Clausthal University of Technology, Julius-Albert-Straße 4, D-38678 Clausthal-Zellerfeld

SNE 30(3), 2020, 117-124, DOI: 10.11128/sne.30.tn.10524
Received: August 8, 2020 (Selected ASIM vSST 2020 Postconf. Publ.); Revised: August 15, 2020; Accepted: August 20, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Production systems in Industry 4.0 are characterized by a high degree of system networking and adaptability. They are often characterized by jointed-arm robots, which have a high degree of adaptation. Networking and adaptivity increase the flexibility of a system, but also the complexity of the control, which requires the use of new development methods. In this context, the Simulation-Based Control approach, a model-based design method, and the concept of Reinforcement Learning (RL) are introduced and it is shown how a task-based robot control can be learned and executed. Afterwards, the time complexity of the Q-learning method will be examined using the application example of a robot-based assembly cell with two differently flexible system configurations. It is shown that, depending on the system configuration, the time complexity of learning can be significantly reduced when using several agents. In the studied case, the complexity decreases from exponential to linear. The modified RL structure is discussed in detail.

Introduction

Production systems of Industry 4.0 have a high degree of networking and adaptivity. The latter characterizes the flexibility of a system to adapt to changing influences [1]. Systems are often characterized by jointed-arm robots which, according to [2], have a high degree

of flexibility in terms of design and control. Adaptivity and networking increase the complexity of the control software, which requires the application of new development methods. In recent years, similar methods have been established under various terms, such as *Model-Based Design (MBD)* [3], *Rapid Control Prototyping (RCP)* [4] or *Virtual Commissioning (VC)* [5]. What they have in common is that they are based on a continuous model- and simulation-based development from the design to the operation phase. The *Simulation-Based-Control (SBC)* approach [6] was developed adequately for this purpose by the Computational Engineering and Automation research group at the University of Applied Sciences in Wismar. This approach was adapted in [7] and [8] specifically for task-oriented control development for jointed-arm robots.

In [9], a connection of the SBC approach with machine learning methods based on *Reinforcement Learning (RL)* according to [10] is shown. The RL is defined by a structure of at least one agent, which has a learning method and an environment. The specific learning method is Q-learning. Based on predefined tasks and transformation modules, a control strategy is learned and automatically transformed into an SBC-compliant program. Since Q-learning is a model-free algorithm, it can be applied to various problems. However, impracticable computing times result relatively quickly, even though learning can often be accelerated for problems with limited state space by means of parallel processing and binary trees [11]. A significant reduction of the state space and, thus, the computational effort is achieved with model-based RL approaches [12]. Here the agent already has process-relevant knowledge at the beginning of the learning process, but thereby loses its universality

ty. For problems with large state space, the computing time can be reduced by combining *artificial neural networks (ANN)* as function approximators [13], [14]. However, this increases the complexity of the software architecture. Furthermore, the design and training of the ANN requires experience and time. In contrast, the simple architecture of the original Q-learning is an advantage if the computational effort can be mastered.

In this paper, we investigate how the computing time for Q-learning of a task-based robot control can be reduced by using several agents. Two differently flexible system structures of an assembly cell are considered and the time complexity of learning with one agent compared to several agents is analyzed. The learning is performed on simulated system environments. The generation of an executable robot control based on the SBC approach and basics of the RL approach using Q-learning are discussed in the following background section.

1 Background

Starting from the adapted SBC framework for articulated arm robot systems according to [7] and [8], this section deals with the principle of integration with a machine learning process and the basics of RL based on Q-learning according to [10] and [13].

1.1 The SBC Approach

As shown in Figure 1, an SBC-based robot control is layered in analogy to the concept of the *Robot Operating System (ROS)* [15].

The *Control Model (CM)* specifies the control strategy by composing predefined basic tasks. The *Process Model (PM)* implements the task transformer based on predefined task-specific modules. Additionally, the PM maps the state information. The *Interface Model (IM)* implements the interface to the hardware using a robot middleware. With the *RCV Toolbox for MATLAB* according to [16] as middleware, SBC-based robot controls can be developed and operated independently of robot manufacturers and model-based with *MATLAB/Simulink*. Virtual and real robots can additionally interact with a virtual process environment in the form of a simulation model.

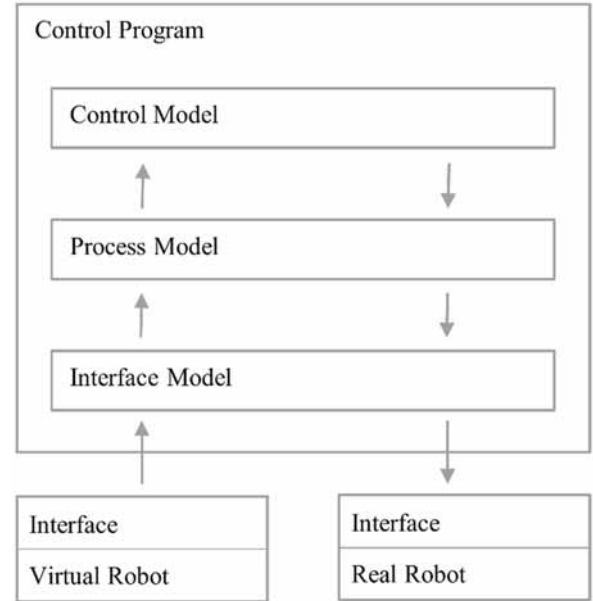


Figure 1: Structure of an SBC-based robot control.

With the integration of the SBC framework with a RL procedure, as introduced in [9], the task-based control specification of the CM according to formula 2 is automatically generated from a sequence of learned state/action tuples according to formula 1:

$$[(s_0, a_0), \dots, (s_t, a_k), \dots, (s_{target}, cancel)] \quad (1)$$

with $s_t \in S, a_k \in A, S$ state set, A action set

$$[move(\dots), pick(\dots), \dots, stop(\dots)] \quad (2)$$

Learning takes place offline using a simulated process environment. In principle, the learning algorithm could also run during the operating phase and adapt the control strategy in certain time windows if the real-time requirements are met.

1.2 Reinforcement Learning with Q-learning method

Besides supervised and unsupervised learning, the RL method forms a third class of machine learning procedures. The goal of RL is to learn a behavioral strategy $\pi: S \rightarrow A$ that assigns an action $a \in A$ to each state $s \in S$. The RL does not require explicit training data. It trains itself using a real or simulated environment according to the *trial and error principle*.

The RL is based on a framework as shown in Figure 2. In model-free RL, the agent only knows the allowed action set A at the start of training. The environment is defined by different states, $s \in S$. When an action $a \in A$ takes effect, the environment determines a subsequent state s' with the state transition model $T: S \times A \rightarrow S$ and computes a reward value $r \in R$ for the current action. The subsequent state s' and the reward r are sent back to the agent. During training, the agent receives information about the possible states of the environment and the benefits of actions through iterative interaction, and gradually learns a behavioral strategy π . After completion of all training episodes, the behavioral strategy π is derived from the Q-matrix.

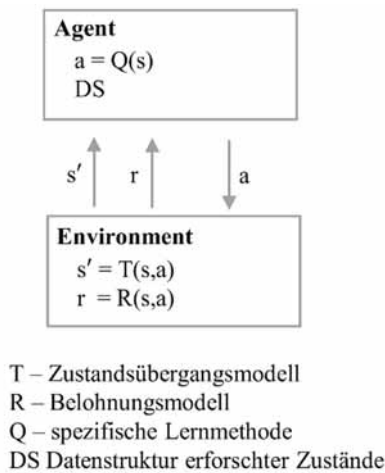


Figure 2: RL framework with Q-learning method.

Learning takes place in phases, also known as episodes. These are independent of each other, always start in an initial state s_0 of the environment and end when a target state s_{target} or abort state s_{abort} is reached. At the beginning of the training, the agent selects an action $a \in A$ purely randomly (*exploration*). As the learning process progresses, the agent increasingly uses the knowledge it has acquired to select an action (*exploitation*). The ratio of exploration to exploitation is adjusted in the course of training.

Q-learning is based on a table function called Q-matrix. A matrix element $Q(s, a)$ represents the benefit Q of an action a when it is performed in the state s of the environment. From the cardinality $|A|$ follows the column dimension of Q . The row dimension of Q grows dynamically during the training with the number of states explored.

The data of explored states are stored in an indexed data structure DS based on the row index of Q . This allows the agent to clearly recognize already explored states. The first episode of the training starts with an empty Q-matrix and an empty data structure. Subsequent episodes build on the already acquired knowledge in Q and DS . After each interaction with the environment, the agent checks whether the received state s' is known. If not, the Q-matrix is extended by a new row and the state data are added to the data structure DS . The successive adjustment of the Q-values results from formula 3 according to [13]:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right] \quad (3)$$

The updated Q-value of the current state/action tuple (s, a) is calculated from the previous Q-value, the currently received reward r , and the maximum Q-value of all possible actions in the currently received subsequent state s' . The influence of the individual variables is determined by the hyperparameters: (i) discount factor γ and (ii) learning rate α . The discount factor controls the influence of rewards expected in the future and the learning rate controls the influence of the current observation. In environments with deterministic behavior, a high learning rate can be applied with up to $\alpha = 1$.

2 Application Example

An automated assembly cell (AC) with an articulated arm robot for the production of different assemblies is considered as an example. By means of RL, product-compliant assembly sequences are to be learned, on the basis of which robot controls can be generated according to Section 1.1. Figure 3 shows the system layout of the AC. The AC consists of the articulated arm robot (R), an assembly station (AS) and a belt conveyor (BC). The BC feeds input parts from upstream production sections and serves as a transfer zone (TZ) to the robot. Depending on the specific system configuration, different numbers of transfer locations (TL) are possible, which influences the system flexibility. Unused input parts and assembled modules are automatically removed. The sequence in which the individual parts enter the AC is unknown and is assumed to be random.

Figure 4 shows an example of an assembled module as an exploded view, whose assembly is examined below for two system variants of the AC:

1. Minimum variant with only one TL as TZ and
2. Variant with three TLs as TZ.

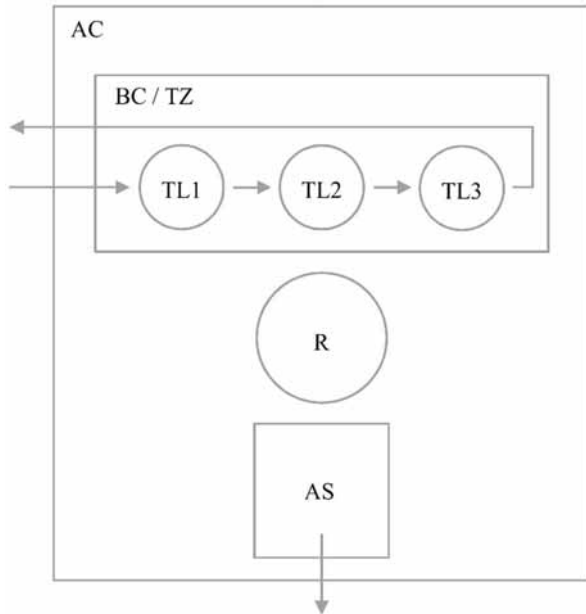


Figure 3: System layout of the assembly cell with maximum three TLs as TZ.

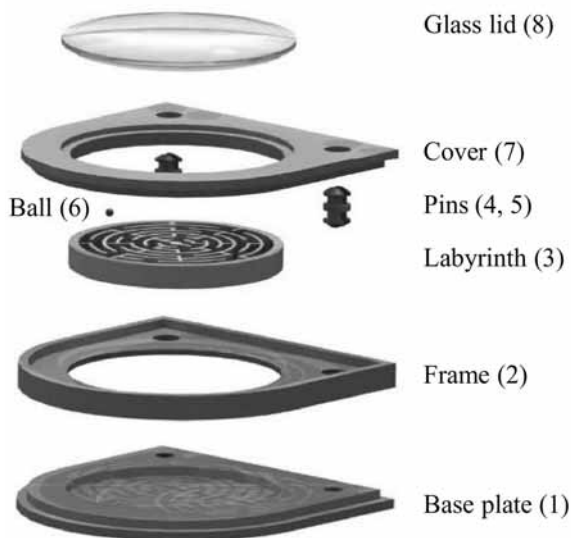


Figure 4: Exploded view of an assembled module according to [7].

For the desired complexity consideration, the assembly is reduced to a *pick & place application*. The assembly of the module in Figure 4 is subject to six rules. Four of the rules are shown in [7]. Rules 5 and 6 were added:

1. Base plate (1) must be mounted first.
2. Pins (4, 5) may only be mounted after cover (7).
3. Comprehensive parts must be mounted after enclosed parts.
4. The glass cover (8) must be mounted last.
5. Each part is only assembled once and the pins (4, 5) are not identical.
6. Gripping attempts on empty TLs of the TZ are prohibited.

In the following two sections, the learning of an assembly strategy with first one agent for both system variants and then using several agents for the second variant is examined.

3 Learning with one Agent

According to Section 1.2, the RL framework consists of an agent and an environment. In the application case under study, the robot forms the agent and the BC and AS form the environment. In this section, the learning of an assembly strategy based on this RL framework is examined for the two system variants of the AC.

3.1 System variant with one TL as TZ

In the case of only one TL, the robot has two possible actions $A = \{0, 1\}$. The action $a = 0$ encodes the task *None* and the action $a = 1$ encodes the task *Pick&Place*. The state of the environment results from the TZ allocation and the assembly state of the module on the AS. In the case of only one TL, there are nine possible states of the TZ with $S_{TZ} = \{0, 1, \dots, 8\}$. The state value 0 stands for no component and the values greater than zero for a component according to the part numbers in Figure 4. The final assembled module consists of eight individual parts. Accordingly, the assembly state on the AS can be represented by an 8-tuple. Each tuple element describes the non-assembly or assembly of a component by the values 0, 1, ..., 8.

A state $s \in S$ of the entire environment is, therefore, described by a 9-tuple. The first eight elements describe the assembly state on the AS and the ninth element the state of the TZ. Figure 5 shows the representation of states $s \in S$ of the environment with a state vector (SV).

State s_0 represents the initial state, s' a possible later subsequent state and s_{target} the target state. The indicated state s' encodes an assembly of the components' base plate and labyrinth as well as the allocation of the TZ with a component glass lid. The target state s_{target} is reached when the first eight elements of SV are not equal to zero. The character X in the initial and target state stands for any state $s_{TZ} \in S_{TZ}$ of the TZ.

Initial state $s_0 \in S$								
0	0	0	0	0	0	0	0	X

A possible subsequent state $s' \in S$								
1	0	3	0	0	0	0	0	8

Target state $s_{target} \in S$								
1	2	3	4	5	6	7	8	X

Figure 5: Mapping of the state as a state vector (SV).

The environment reacts to an action a of the agent, as shown in Figure 2, with a reward value r and a subsequent state s' . The reward model R of the environment defines three possible rewards for an action $a = 1$ based on the six assembly rules according to Section 2:

- $r = -\infty$, if the action is not allowed,
- $r = 0$, if the action is allowed and $s' \neq s_{target}$ and
- $r = 1$, if the action is allowed and $s' = s_{target}$.

The action $a = 0$, i.e. a refusal to mount the component on the TL, always leads to the reward value $r = 0$.

The state transition model T works according to the Markov Decision Process (MDP) paradigm. Listing 1 shows the specification of T in MATLAB pseudocode.

```
%A={0,1} action set of the agent
1 num_parts = 8;
2 idx = num_parts + 1;
3 a = action(agent); %current action
%Pick&Place only
4 if a > 0
5     SV(SV(idx)) = SV(idx);
6     SV(idx) = 0;
7 end
%Pick&Place and None
8 SV(idx) = randi(num_parts + 1) - 1;
```

Listing 1: State transition model for an TZ with one TL.

With a permitted action $a = 1$ (*Pick&Place*), the last element of the SV that represents the part in the TC is restored according to its value in the SV (lines 4-7), thereby updating the assembly state. In addition, for each allowed action $a = 1$ or $a = 0$ the last element of SV is assigned a random integer value from the interval $[0, num_parts]$, which represents a new input part in the TC (line 8).

Each episode starts with the transfer of a start state s_0 of the environment to the agent. A reward is not transferred at the beginning of an episode. Learning is done as described in Section 1.2. The column dimension of the Q-matrix corresponds to the cardinality of the action set $|A| = 2$. An episode ends when the agent receives a reward, $r = 1$, from the environment. The training must include enough episodes to learn a strategy, π . It should be noted that, with the random generation of new input parts in the TZ, the state transition model T is subject to stochastic influences.

Without assembly rules, this system structure results in $2^n(n+1) - n$ states of the environment depending on the number of parts n to be assembled. Due to assembly rules, the number of states is considerably reduced. For the assembly under consideration with eight components, 144 states result according to [7].

The complexity of an algorithm as a function of the input data is described with the big O notation [17]. An empirical analysis showed that a linear time complexity $O(n)$ of the learning process resulted depending on the number of components, n , to be assembled. The computing time for learning an assembly strategy was less than one hour on a standard PC with an implementation in MATLAB.

3.2 System variant with three TLs as TZ

In the variant with three TLs the allocation of the TZ is analogous to a shift register. In fixed time units, the BC moves one position to the right. The robot always has access to all three TLs. This results in four possible actions for the robot, $A = \{0,1,2,3\}$. The action $a = 0$ again encodes the task *None* and the other three actions encode the task *Pick&Place* with different parameterization depending on the TL to be approached.

The state of the environment expands by two elements to an 11-tuple. The first eight elements describe the assembly state on the AS. The other three elements each describe the non-occupancy or occupancy of a TL.

Due to the three TLs, the number of possible states of the environment increases. For the TZ with three TLs and eight possible input parts, as well as the case of non-occupancy of a TL, results in 729 states $s_{TZ} \in S_{TZ}$ with $S_{TZ} = \{(0,0,0), (1,0,0), \dots, (5,1,1)\}$. The values 1 to 8 encode a component according to the part numbers in Figure 4 and the value 0 encodes the non-occupancy of a TL.

The calculation of rewards $r \in R$ is analogous to the variant with only one TL. The state transition model T , which has been extended to a TZ with three TLs, shows Listing 2.

```
%A={0,1,2,3} action set of the agent
1 num_parts = 8;
2 a = action(Agent); //current action
3 idx = a + num_parts;
//Pick&Place only
4 if a > 0
5   SV(SV(idx)) = SV(idx);
6   SV(idx) = 0;
7 end
%Pick&Place and None
%Move TL allocations (TL1→TL2→TL3)
%cardinality |A| is 4
8 for k = |A| - 1 : -1 : 2
9   idx = k + num_parts;
10  SV(idx) = SV(idx - 1)
11 end
%generate new input part on TL1
12 idx = num_parts + 1;
13 SV(idx) = rand_i(num_parts + 1) - 1;
```

Listing 2: State transition model for the TZ with three TLs.

The 11-tuple representing the state of the environment is again implemented as a state vector (SV). If the agent selects a permitted action $a = \{1,2,3\}$ for a *Pick&Place* task (line 2), the input part is taken from the corresponding TL in the TZ and mounted at the AS by the robot (line 4-7). Subsequently, for each allowed action $a \in \{1,2,3\}$ of the robot, the shift register movement is realized by the BC (lines 8-11) and a new input part is randomly generated in the form of an integer value in the interval $[0, \text{num_parts}]$ for the first TL in the TZ (lines 12-13).

Learning a behavioral strategy π is analogous to the representation in Section 3.1. However, despite the applicable assembly rules, the state space of the environment increases exponentially with $144 \cdot 9 \cdot 9 = 11644$ states.

The complexity of the learning algorithm corresponds to $O(n^c)$ with n the number of components to be assembled and c the number of TLs in the TZ. In an empirical study, the computing time was about 10 hours and it was about ten times longer than the computing time for the variant with only one TL.

4 Learning with Multiple Agents

Based on the structure of the application problem, a multi-agent approach to reduce the computing time appears to be appropriate for the second system variant with three TLs in the TZ. Figure 6 shows an RL framework consisting of three agents, an environment, and a *management* component.

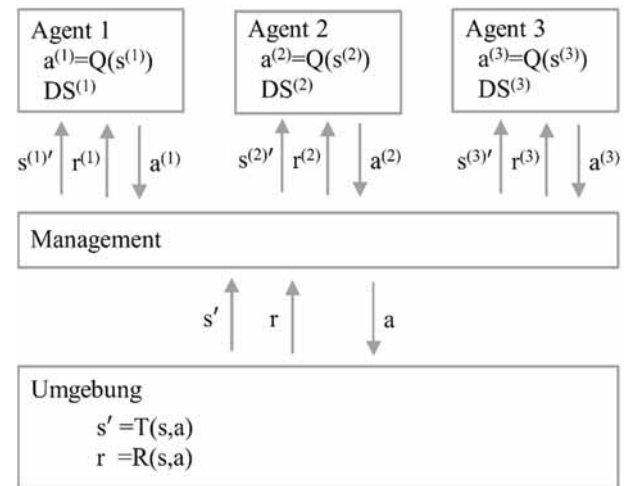


Figure 6: RL framework with multiple agents.

The three agents are identical. Their structure and behavior correspond to the individual agent in Section 3.1, and their action set $A = \{0,1\}$ represents the two tasks *None* and *Pick&Place*. The models T and R of the environment correspond to the representation in Section 3.2. Accordingly, the environment reacts to the action set $A = \{0,1,2,3\}$. In contrast to the original RL framework, the agents and the environment do not communicate directly with each other, but via the intermediate management. The management decomposes each state s' as shown in Figure 7. The first eight elements of s' , which encode the assembly state on the AS, are passed on to all three agents. From the ninth to the tenth element, which code the allocation of the three TLs in the TC, each agent only receives the state of one TL at a time.

This reduces the state space from the viewpoint of each individual agent to 144 states, analogous to the agent in Section 3.1.

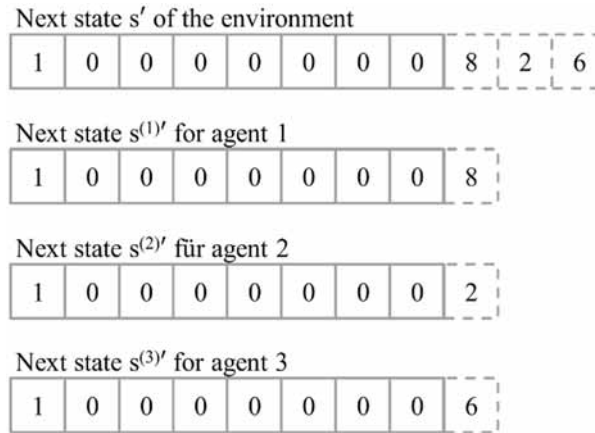


Figure 7: Decomposition of a next state s' of the environment by the management.

An episode begins with the decomposition of the initial state s_0 of the environment by the management, analogous to Figure 7. Then, the management sends to each agent i the corresponding reduced state vector $s^{(i)'}$, as shown in Figure 6. Each agent reacts individually with an action $a^{(i)} \in \{0,1\}$. The management selects one of the three actions according to the following criteria:

1. An action $a = 1$ (*Pick&Place*) is preferred over an action $a = 0$ (*None*).
2. If several agents send an action $a = 1$, the agent with the highest index i is selected. The prioritization of the agents follows from the way the TZ works as a shift register. Parts on the second and first TL are still available in the next and next but one cycle. According to Figure 7, *Agent 1* has the lowest priority and *Agent 3* the highest.

The management remembers the selected agent. If an action $a = 0$ was selected, the action is forwarded to the environment unchanged. If an action $a = 1$ was selected, the action is converted into a value of the action set $A = \{1,2,3\}$ based on the index i of the agent and sent to the environment as action a . The environment reacts to the action with a reward r and a follow-up state s' . These are calculated with the reward model R as well as the state transition model T according to Section 3.2.

The management sends the reward r as $r^{(i)}$ to the agent i , whose action was selected before. The other agents receive an empty reward value. An empty reward means that there is no learning for the agent in this iteration step because its action has been discarded. The subsequent state s' is decomposed by management as described and the reduced states $s^{(i)'}$ are sent to the agents. Each agent learns a behavior strategy $\pi^{(i)}$ related to the TL assigned to it. Finally, an overall assembly strategy π is derived from the individual strategies.

The number of necessary episodes is hardly different from learning with one agent. Since only one agent is active in each iteration step, the state space to be analyzed has a maximum of 144 states. The empirical investigations revealed a linear complexity $O(n)$ of learning depending on the number n of components to be assembled. The multi-agent approach is scalable regarding the selection of components in the TZ and the complexity is, therefore, independent of the number of TLs. The computing time requirement is reduced to about one hour.

5 Conclusion and Outlook

Starting from the basics of RL using the method of Q-learning and the model-based SBC approach, it was shown how, in principle, a task-based control can be learned and executed.

Subsequently, the learning of a typical pick and place strategy for two differently flexible system structures was considered using the example of a robot-based assembly cell. The focus of the consideration was the time complexity of learning. For the simple system structure with binary selection option, it was found that the learning algorithm has a linear complexity $O(n)$ depending on the number n of assembled components. In contrast, the second system structure with c many simultaneous choices per assembly step had an exponential complexity $O(n^c)$ of learning.

As a result, an RL multi-agent framework with the Q-learning method was designed for the second system structure. It could be shown that, for learning with the multi-agent approach, a linear complexity $O(n)$ results as a function of the number n of components to be assembled and that the complexity is independent of the number of simultaneous choices due to the scalability of the approach.

The application example under study is characterized by learning an assembly sequence that conforms to the assembly. The different system structures were purposefully modelled as environments for the RL according to the MDP paradigm. In subsequent investigations, further influencing variables of a production process are to be taken into account during learning, such as the introduction of input parts as a function of buffer capacities. This requires the integration of the RL with a more complex dynamic simulation model of the production, which was not explicitly developed according to the MDP paradigm. Conceptual approaches for such simulation-based RL experiments are presented in Schmidt [18] and Adams [19].

References

- [1] Bundesministerium für Wirtschaft und Energie. Was ist Industrie 4.0? (Federal Ministry for Economic Affairs and Energy. What is Industry 4.0?) <https://www.plattform-i40.de/PI40/Navigation/DE/Industrie40/WasIndustrie40/was-ist-industrie-40.html> [Retrieved 27-July-2020].
- [2] Hägele M., Nilsson K., Pires J.N. Industrial Robotics. In: Siciliano B., Khatib O., editors. Handbook of Robotics. Berlin: Springer Pub; 2008. 963-986.
- [3] Nicolescu G., Mosterman P.J. Model-Based Design for Embedded Systems. Boca Raton / FL: CRC Press; 2010. 766 p.
- [4] Abel D., Bollig A. Rapid Control Prototyping – Methoden und Anwendungen (Methods and Applications). Berlin: Springer Pub., 2006, 400 p.
- [5] Turnbull C. What is Virtual Commissioning? <https://virtualcommissioning.com/what-is-virtual-commissioning/> [Retrieved 27-July-2020].
- [6] Pawletta T., Pawletta S., Maletzki G.: Integrated Modeling, Simulation and Operation of High Flexible Discrete Event Controls. In I. Troch, F. Breiteneker, editors, Proc. Mathematical Modelling - MATHMOD 2009 Feb, Vienna. Argesim Report No. 35, 13 p., ISBN 978-3-901608-35-3
- [7] Maletzki, G. Rapid Control Prototyping komplexer und flexibler Robotersteuerungen auf Basis des SBC-Ansatzes (Rapid control prototyping of complex and flexible robot controls based on the SBC approach) [Dissertation]. Universität Rostock / Hochschule Wismar; 2013. In: ASIM FBS 25 doi: 10.11128/fbs.25.
- [8] Freymann, B. Aufgabenorientierte Multi-Robotersteuerungen auf Basis des SBC-Frameworks und DEVS (Task-oriented multi-robot controls based on the SBC framework and DEVS) [Draft Dissertation]. TU Clausthal / Hochschule Wismar; 2020 (unpublished).
- [9] Kunert G. Pawletta T. Generating of Task-Based Controls for Joint-Arm Robots with Simulation-based Reinforcement Learning. SNE – Simulation Notes Europe. 2018; 28(4):149-156. doi:10.11128/sne.28.4.1044
- [10] Sutton R., Barto A. Reinforcement Learning. 2nd Edition. Cambridge/ MA: MIT Press; 2012. 334 p.
- [11] Jammer D., Pawletta S., Kunert G., Pawletta T. Beschleunigung eines Reinforcement-Learning-Algorithmus durch Parallelverarbeitung für Robotikanwendungen (Accelerate a reinforcement learning algorithm through parallel processing for robotics applications.). In Durak U. et al., editors. Proc. ASIM STS/GMMS Symposium; 2019 Feb; Braunschweig. Wien: ARGESIM Verlag. 49-52. doi: 10.11128/arep.57.
- [12] The MathWorks. Reinforcement Learning With MATLAB – Part 1. Ebook. The MathWorks Inc.; 2019. 24 p.
- [13] Russel S., Norvig P. Artificial Intelligence: A Modern Approach. 4nd Edition. Cambridge / MA: MIT Press; 2020. 1115 p.
- [14] Zai A., Brown B. Deep Reinforcement Learning in Action. Shelter Island / NY: Manning Pub.; 2020. 359 p.
- [15] ROS-Industrial. Homepage. <https://rosindustrial.org> [Retrieved 13-May-2020].
- [16] Deatcu, C., Freymann, B., Schmidt, A., Pawletta, T. MATLAB/Simulink Based Rapid Control Prototyping for Multivendor Robot Applications. SNE – Simulation News Europe. 2015; 25(2): 69-78. doi:10.11128/sne.25.2.1029.
- [17] Filho W.F. Computer Science Distilled. Las Vegas / NV: Code Energy LLC Pub.; 168 p.
- [18] Pawletta T., Durak U., Schmidt A. Modeling and Simulation of Versatile and Adaptable Systems with an Application in Engineering. In Zhang L. et al., editors, Model Engineering for Simulation. Elsevier Inc. Pub., 2019, Chap. 18, 29 p.
- [19] Adams S. et al. Reinforcement Learning from Simulated Environments: An Encoder Decoder Framework. In Proc. SCS SpringSim'20, 2020 May 19-21, Fairfax / VA, 12 p.

Modeling and Simulation Using Cellular Automata

Oliver Ullrich*, Daniel Lückerath

Fraunhofer Institute for Intelligent Analysis and Information Systems, Schloss Birlinghoven,
53757 Sankt Augustin, Germany *oliver.ullrich@iais.fraunhofer.de

SNE 30(3), 2020, 125-130, DOI: 10.11128/sne.30.en.10526
Received: July 1, 2020; Revised: August 18, 2020;
Accepted: August 20, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Cellular Automata are a simple, yet powerful modeling and simulation technique, easily employed if the system to be examined consists of a set of uniform entities that are located in a neighborhood relation to each other. The simple structure and fast execution of Cellular Automata allow for very large models, their inherent parallelism enables a comparatively simple parallelization.

This paper gives an overview on the concepts of Cellular Automata-based modeling and simulation, with a special focus on neighborhood concepts, border types, and transition functions, and describes current applications of the paradigm in the areas of the simulation of transportation networks, additive manufacturing, as well as urban growth and development. It is aimed at students of the craft as well as practitioners who might want to take a look beyond the GUI of their modeling tools.

Introduction

Beyond their application in simulation modelling, Cellular Automata (CA) are also studied in other areas of computer science, in mathematics, biology, and physics. CA generally consist of a regular grid of homogenous cells, each of which is assigned one of a finite set of states. A cell changes its state over time based on a transition function – which in turn is determined by the states of a set of other cells, the original cell's neighborhood.

The core concepts of CA were developed in the 1960s by John von Neumann (see [27]) as a tool to formally model self-reproducing biological systems (see [24]). In the 1970s they were developed further, with the aim to examine what was then called “artificial life”

– the best known artifact of that period probably is Conway's Game of Life (see [6]). It was shown that CA are capable of self-reproduction in a way that is sufficient to support a universal computer – they are thus Turing complete (see [24]).

While CA today are applied in a number of domains, for example fault-tolerant quantum computing (see [25]), image encryption (see [19]), modeling of epidemic spreading (see [17]), and object detection in computer vision (see [21]), this paper takes a closer look at their application in the area of modeling and simulation.

In many application fields in the simulation area, CA have been superseded by agent-based modeling and simulation. However, some advantages remain: their simple structure and extremely fast execution allow for very large models, their inherent parallelism enables a comparatively simple parallelization. CA can also be used for high-performance simulation of environments for agent-based models.

This paper gives an introduction on the use of CA in modeling and simulation; it explains both the fundamentals of the technique as well as its applications during the last few years.

The paper continues with a short recap of the basics of simulation modeling (see Section 1), followed by an introduction of CA with a focus on neighborhood concepts and transition functions (see Section 2). It then reports on current applications in modeling and simulation, specifically in the areas of transportation networks, additive manufacturing processes, and urban development simulation (see Section 3). The paper concludes with a short summary of the lessons learned (see Section 4).

1 Simulation Modeling

The term simulation can cover a wide variety of processes and models. In its broadest definition, simulation simply refers to the performance of experiments

on a model in order to make statements about the behaviour of a real system or process (see [16]). A narrower definition of simulation, which is slightly adapted and also used for this work, is formulated by Ingalls (see [8]): Simulation is the performance of experiments on a dynamic and stochastic model with the aim of making statements about the behaviour of a real dynamic and stochastic system or process. *Dynamic* in this context means that the behavior of the system – and thus of the model – changes over time, while *stochastic* means that the behavior can be influenced by random variables.

This allows simulation models to be distinguished from (mathematical) analytical models, such as optimization models. Analytical models try to formulate correlations of real systems as sets of mathematical equations. This formulation usually requires the definition of rigid specifications, for example fixed travel times between stops in transit models, in order to obtain manageable models. If, however, the system under consideration meets the requirements, the application of the analytical model always leads to a valid solution. Simulation models, on the other hand, attempt to model the behaviour of the real system as complex behavioral and decision-making processes. This may no longer guarantee that the solution obtained represents an optimum, but allows to represent dynamic and stochastic behaviour (see [12]).

The group of dynamic and stochastic systems includes, for example, technical systems and transportation systems – simulation models are indispensable research tools in those areas. Typically, discrete simulation methods are used to represent such systems, for example CA-based, event-based (see [26]), or agent-based (see [15]) simulation. These paradigms differ in the specific way they represent system components and their interdependencies, as well as in the specific representation of concepts such as time progress. In any case the modeler has to decide which system components and interrelations are critical for representing the behaviour of interest, and which modeling methods are most appropriate (see [12]).

2 Cellular Automata

If the system to be simulated consists of a set of uniform entities that are located in a (geographical) neighborhood relation to each other, Cellular Automata can be employed easily and efficiently.

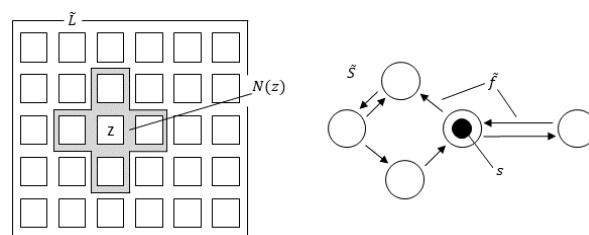


Figure 1: Cellular Automaton $Z = (\tilde{L}, \tilde{S}, N, \tilde{f})$ consisting of a regular, two-dimensional grid \tilde{L} of square-shaped cells $z \in \tilde{L}$ with state s from the set of states \tilde{S} and with a transition function \tilde{f} that is dependent on a neighborhood $N(z)$.

2.1 Introduction

A Cellular Automaton $Z = (\tilde{L}, \tilde{S}, N, \tilde{f})$ is a (infinite) regular grid \tilde{L} of cells $z \in \tilde{L}$, each of which may be in any state s from the finite set of states \tilde{S} (see Figure 1). The state of a cell changes over time based on a transition function \tilde{f} . Each cell z forms a neighborhood with a finite set of other cells $N(z) = \{i \in \tilde{L} | z - i \in N\}$. The characteristics of that neighborhood are determined by the set of permissible neighborhood indices N . This set of indices in turn depends primarily on the spatial discretization of the model. In principle, such a lattice geometry can have any number of dimensions; typically, in practical applications one, two, or three dimensional lattices are used. The lattice geometry also defines the shape of the cells. Any cell shape that guarantees the regularity of the grid is conceivable; frequently encountered cell shapes are: square, triangular, or hexagonal. For a more detailed description of these concepts see [28].

2.2 Neighborhood Concepts

Figure 2 shows the frequently encountered von Neumann and Moore neighborhood concepts in the context of a number of lattice geometries. As can be seen, the cells of a von Neumann neighborhood of a cell z share a whole edge of the lattice with z . The Moore neighborhood, on the other hand, also includes all cells that have only one corner point in common with z .

Depending on the characteristics of the system to be modeled, the neighborhood of a cell may not necessarily include only direct neighbors – i.e. cells with distance 1 – but may also include cells further away. That might be useful, if obstacles further away, for example

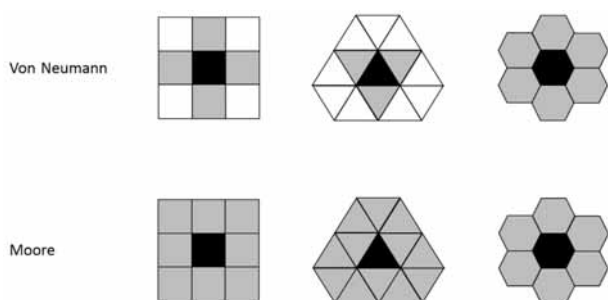


Figure 2: Two often used neighborhood concepts: Von Neumann neighborhoods with shared edges and Moore neighborhoods with shared edges and corner points.

upcoming traffic signals in a public transit simulation, have to be taken into account.

In some models, it is beneficial to define distinct input and output neighborhoods: a cell takes its input from its input neighborhood, while making its state available to the cells of its output neighborhood (see [24]).

For practical applications, the maximum expansion of the grid is usually limited, which makes a specific consideration of border cells necessary. Usually, three different types of border cells are distinguished:

- With *periodic borders*, opposite sides of the grid are adjacent;
- In case of *reflecting borders* the border cells reflect the state of the inner cell opposite them; and
- With *fixed state borders*, the border cells are assigned a fixed state each, which does not change over the course of the simulation.

The grid boundary does not necessarily have to consist of a single border type, but can combine different types.

2.3 Transition Functions

When modeling the lattice geometry, decisions have to be made regarding the time progress and the transition function of the CA. In CA the simulation time progresses in fixed increments δt between discrete points in time. At each simulation time t the configuration $\tilde{C}_t : \tilde{L} \rightarrow \tilde{S}$ assigns a current state to each cell of the grid. The configuration \tilde{C}_0 designates the start configuration of the CA specified by the modeler. To update

the model state over simulation time, a transition function $\tilde{f} : S^{|\tilde{N}|} \rightarrow \tilde{S}$ transfers the configuration \tilde{C}_t into the configuration \tilde{C}_{t+1} as shown in Equation 1.

$$\tilde{C}_{t+1}(z) = \tilde{f}(\{\tilde{C}_t(i) | i \in N(z)\}) \quad \forall z \in \tilde{L} \quad (1)$$

The transition function is the model characteristic that pivotally determines the behavior of the CA. It depends on the lattice geometry, the type and size of the neighborhood, as well as the number of possible states; its specific design has to be determined together with these model components based on the objectives of the investigation.

Generally, two types of transition functions exist: In a *deterministic transition function*, state transitions are directly determined by the state of the affected cell and the states of its neighbors. With *probabilistic transition functions*, the state transition to be applied in each simulation time step is randomly selected from a set of potential state transitions.

Due to the close relationship between lattice geometry, neighborhood, and transition function, spatial and temporal discretization are directly related in CA. Specifically, a decision on the cell dimensions has a direct influence on the size of the time increment δt . If, for example, a cell length of 30 meters – about one rail car length – is selected for the simulation of light rail systems, a vehicle entity can only move by multiples of these 30 meters within one simulation step. The duration of δt has to be assigned accordingly. As a result of these considerations, the duration of one simulation step might be set to correspond to four seconds in the real system (see [10] and [14]), so that vehicles moving at velocities of 0, 27, 54, and 81 kilometers per hour can be represented.

The tight interdependencies between spatial and temporal dimensions do not arise with event-based and agent-based simulation methods, and are one main disadvantage of CA models.

3 Applications

CA-based modeling and simulation is utilized in a large variety of subjects. In the following, three areas are examined: the simulation of transportation networks, where CA-based explanatory and predictive models are employed since the 1990s; additive manufacturing, where simulation models help to understand how metal

powders are melted by a focused laser beam; and urban growth, where simulation studies help to predict how metropolitan regions might develop and how climate change-related effects will impact them.

3.1 Simulation of Transportation Networks

One-dimensional CA seem to be a natural choice to model linear transportation infrastructure, for example roads or railways. The CA-based examination of transportation phenomena started out with Nagel and Schreckenberg (see [18]) in 1992. They used a simple CA model with periodic borders representing a circular roadway to explain the development of spontaneous congestion forming without an external cause, for example an accident or construction site. They showed that such congestion can form on roads with high-density traffic when cars slow down to avoid colliding with the preceding vehicle. Resulting from a driver's non-zero reaction time when the preceding car eventually accelerates, a ripple effect emerges that slowly moves backwards through the stream of succeeding cars.

The Nagel/Schreckenberg model has continuously been extended and repurposed. In a recent study, Iwan et al. (see [9]) extend the basic configuration to estimate the environmental impact of unloading bays in city centers. They state that such unloading bays are a simple and effective solution to reduce the congestion resulting from urban freight transport and support that argument with results from CA-based experiments. Their experiments with models of the city centers of Szczecin, Poland and Oslo, Norway show a significant impact of the integration of unloading bays on reducing negative environmental effects of urban freight transport: In time periods with a high ratio of freight vehicles – typically in the mornings – pollutants can be reduced by 15-18 percent by providing adequate unloading bays.

In another road traffic-related application, Wu et al. (see [29]) examine a number of warning systems aimed at helping drivers to decide on whether to stop or go when a green traffic light turns to red right in front of them. They construct a set of scenarios regarding combinations of warning measures and devise a CA-based model to evaluate them. They find that a combination of pavement markings and flashing yellow auxiliary lights consistently provide the lowest probability of rear end crashes.

In addition to examining road traffic phenomena, a tradition exists of examining ship traffic in harbors and

waterways using CA simulation models. Liu, Zhou, and Wang (see [13]) start out with a rather basic simulation model of the traffic flow in single-lane waterways, with Feng (see [4]) adding the representation of length and velocity of different ship types, as well as multi-lane and bottlenecked waterways. Qi, Zheng, and Gang (see [20]) add a more complex system of spatial representation and find that their model maps the ship traffic in the Singapore Strait satisfactorily.

3.2 Simulation of Additive Manufacturing Processes

Additive manufacturing (AM) is a set of relatively new technologies to individually fabricate complex parts without expensive individual tooling or set-up cost. One method of AM is selective laser melting, where thin layers of metal powder are one by one melted by a focused laser beam. In this area, understanding and predicting microstructures of manufactured materials and their behavior has seen significant research focus. Both two-dimensional and three-dimensional CAs are applied here, the former to simulate individual layers, the latter to examine the whole structure.

Zinoviev et al. (see [30]) develop a two-dimensional, CA-based model that provides a realistic prediction of the evolution of layers of grain structure produced by multiple beam passes during laser melting. Their simulation model yields typical phenomena found in real-world data, for example grain growth competition, through-the-thickness grain growth, and texture formation. They note that, to fully explore the characteristics of the built-up of grain structure, a 3D model would have to be developed.

Rai, Markl, and Körner (see [22]) start out on another approach to simulate grain structure evolution based on a combined 2D CA–Lattice Boltzmann model. Their complex model represents a set of major aspects of AM, including random powder layer generation, electron beam energy absorption, evaporation, capillarity and wetting, meltpool dynamics, temperature evolution, and grain solidification. Akram et al. (see [1]) specifically examine the solidification texture of melted metal powder. Understanding that phenomenon can provide a pathway to better understand and control these features, leading to a stream-lined process with better results.

Taking the step from examining individual layers to modeling the whole AM process, Zinovieva, Zinoviev, and Ploshikhin (see [31]) develop a 3D model com-

binning CA with the finite difference method: The CA model is used to simulate the evolution of grain structure in the solidification processes, while the coupled finite difference method describes the thermal processes in the course of the laser passing through the powder. They specifically examine the creation of titanium parts and find the simulation results reproducing the main features of the observed process. They conclude that their model can be applied to understand the variability in the AM process and the resulting titanium parts.

3.3 Simulation of Urban Growth and Development

During the last few years, the examination of urban growth and development using simulation models has seen strong research attention. Two-dimensional CA are obvious candidates for the spatially explicit simulation of land-use and land-cover changes.

Based on urban growth models developed by Clarke, Hoppen, and Gaydos (see [3]) in the 1990s, Sakieh et al. (see [23]) examine cause-effect relationships of urban growth using the example of Karaj, an Iranian city that experienced a substantial increase in urban sprawl during the last three decades. Following on model validation using historical data, they predict urban growth scenarios up to the year 2040, including extensive growth as well as more compact development patterns. They find that, should urban development not be reigned in, the urban land-use in Karaj will grow drastically in the cause of the next two decades. In a similar study, Liao et al. (see [11]) demonstrate the applicability of CA to predict urban growth using the city of Xiamen, China as an example. With good results, Feng, Liu, and Batty (see [5]) apply a more complex CA model to hindcast growth patterns in the Shanghai Qingpu–Songjiang area of China.

To model extreme weather events caused by climate change, urban development models can be augmented by CA-based models of fluvial and pluvial flooding, as described by Guidolin et al. (see [7]).

Moving from conducting individual case studies to preparing meta-analyses and literature reviews, Berberoglu, Akin, and Clarke (see [2]) compare a similar CA model to a number of other methods aimed to assess and predict urban growth based on a model of the city of Adana, Turkey. They compare the simulation results to satellite and aerial images and find that the CA model yields the most exact results. Some weaknesses remain: namely in the quantitative aspect and the in-

ability to include the driving forces of urban growth in the model. They recommend to overcome these weaknesses by combining CA with other quantitative approaches, such as Markov chains and frequency ratio models.

4 Conclusion

This paper presented an overview on Cellular Automata-based modeling and simulation, describing neighborhood concepts, border types, and transition functions. It shared some current applications of the paradigm in the areas of the simulation of transportation networks, additive manufacturing, as well as urban growth and development.

Cellular Automata are an elegant, simple, and powerful modeling and simulation technique, naturally employed if the system to be examined consists of a set of uniform entities located in a one- or more-dimensional neighborhood relation to each other. CA-based simulation models have been applied in many research fields for their explanatory power as well as for their high execution performance.

In the sixth decade since their inception by John von Neumann, researchers and practitioners find new applications and gain new knowledge through the exploration and utilization of Cellular Automata. It should be fascinating to watch what the future will bring for them.

References

- [1] Akram J, Chalavadi P, Pal D, Stucker B. Understanding grain evolution in additive manufacturing through modeling. *Additive Manufacturing*, 21 (2018), pp. 255-268.
- [2] Berberoglu S, Skin A, Clarke K C. Cellular automata modeling approaches to forecast urban growth for adana, Turkey: A comparative approach. *Landscape and Urban Planning*, 53 (2016), pp. 11-27.
- [3] Clarke K C, Hoppen S, Gaydos L. A self-modifying cellular automaton model of historical urbanization in the San Francisco Bay area. *Environment and Planning B: Planning and Design*, 24 (1997), pp 247-261.
- [4] Feng H. Cellular Automata Ship Traffic Flow Model Considering Integrated Bridge System. *International Journal of u- and e-Service, Science and Technology*, 6 (6) (2013), pp. 121-132.
- [5] Feng Y, Liu Y, Batty M. Modeling urban growth with GIS based cellular automata and least squares SVM

- rules: a case study in Qingpu–Songjiang area of Shanghai, China. *Stochastic Environmental Research and Risk Assessment*, 30 (2016), pp. 1387-1400.
- [6] Gardner M. Mathematical Games – The fantastic combinations of John Conway's new solitaire game "life". *Scientific American*, 223 (4) (1970), pp. 120–123.
- [7] Guidolin M, Chen A S, Ghimire B, Keedwell E C, Djordjevic S, Savic D A. A weighted cellular automata 2D inundation model for rapid flood analysis. *Environmental modeling and Software*, 84 (2016), pp. 378-394.
- [8] Ingalls R. Introduction to simulation. *Proc. 40th Winter Simulation Conference*, 2008, pp. 17-26.
- [9] Iwan S, Kijewska K, Johansen B G, Eidhammer O, Malecki K, Konicki W, Thompson R G. Analysis of the environmental impacts of unloading bays based on cellular automata simulation. *Transportation Research Part D: Transport and Environment*, 61(A) (2018), pp. 104-117.
- [10] Joisten M. *Simulation von Fahrplänen für den ÖPNV mittels Zellularautomaten*. Köln, Universität zu Köln, Diplomarbeit, 2002.
- [11] Liao J, Tang L, Shao G, Su X, Chen, D, Xu T. Incorporation of extended neighborhood mechanisms and its impact on urban land-use cellular automata simulations. *Environmental Modelling and Software*, 75 (2016), pp. 163-175.
- [12] Liebl F. *Simulation*. 2nd ed. München: Oldenbourg Verlag, 1995.
- [13] Liu J, Zhou F, Wang M. Simulation of waterway traffic flow at harbor based on the ship behavior and cellular automata. *Proc. 2010 IEEE International Conference on Artificial Intelligence and Computational Intelligence*, 2010, pp. 542-546.
- [14] Lückemeyer G, Speckenmeyer E. Comparing Applicability of Two Simulation Models in Public Transport Simulation. *Proc. 19th Symposium on Simulation Techniques ASIM*, 2006.
- [15] Macal C M, North M J. Introductory Tutorial on Agent-based Modeling and Simulation. *Proc. 2013 Winter Simulation Conference*, 2013, pp 362-376.
- [16] Maria A. Introduction to modeling and simulation. *Proc. 1997 Winter Simulation Conference*, 1997, pp. 7-13.
- [17] Miksch F, Haim C, Schneckenreither, G. Modelling and Simulation of a SIR-type Epidemic with Cellular Automata and Ordinary Differential Equations. *Simulation Notes Europe SNE*, 25(1), 2015, pp. 49-54.
- [18] Nagel K, Schreckenberg M. A cellular automaton model for freeway traffic. *Journal de physique I*, 2, 1992, pp. 2221-2229.
- [19] Niyat A Y, Moattar M H, Torshiz M N. Color image encryption based on hybrid hyper-chaotic system and cellular automata. *Optics and Lasers in Engineering*, 90 (2017), pp. 225-237.
- [20] Qi L, Zheng Z, Gang L. A cellular automaton model for ship traffic flow in waterways. *Physica A: Statistical Mechanics and its Applications*, 471(C) (2017), pp. 705-717.
- [21] Qin Y, Lu H, Xu Y, Wang H. Saliency Detection via Cellular Automata. *Proc. 2015 IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 110-119.
- [22] Rai A, Markl M, Körner C. A coupled Cellular Automaton–Lattice Boltzmann model for grain structure simulation during additive manufacturing. *Computational Materials Science*, 124 (2016), pp. 37-48.
- [23] Sakieh Y, Amiri B J, Danekar A, Fegghi J, Dezhkam S. Simulating urban expansion and scenario prediction using a cellular automata urban growth model, SLEUTH, through a case study of Karaj City, Iran. *Journal of Housing and the Built Environment*, 30 (2015), pp. 591-611.
- [24] Sarkar P. A Brief History of Cellular Automata. *ACM Computing Surveys*, 32 (1) (2000), pp. 80-107.
- [25] Sun M, Lv H, Zhang Y, Xie G. The Fundamental Primitives with Fault-Tolerance in Quantum-Dot Cellular Automata. *Journal of Electronic Testing*, 34 (2) (2018), pp. 109-122.
- [26] Ullrich O, Lückerrath D. An Introduction to Discrete-Event Modeling and Simulation. *Simulation Notes Europe (SNE)*, 27 (1) (2017), pp. 9-16.
- [27] Von Neumann J, Burks A W. *Theory of self-reproducing automata*. Urbana, University of Illinois Press, 1966.
- [28] Weimar J. *Simulation with Cellular Automata*. 2nd ed., Berlin: Logos Verlag, 2003.
- [29] Wu Y, Abdel-Aty M, Ding Y, Jia B, Shi Q, Yan X. Comparison of proposed countermeasures for dilemma zone at signalized intersections based on cellular automata simulations. *Accident Analysis and Prevention*, 116 (2018), pp. 69-78.
- [30] Zinoviev A, Zinovieva O, Ploshikhin V, Romanova V, Balokhonov R. Evolution of grain structure during laser additive manufacturing. Simulation by a cellular automata method. *Materials & Design*, 106 (2016), pp. 321-329.
- [31] Zinovieva O, Zinoviev A, Ploshikhin, V. Three-dimensional modeling of the microstructure evolution during metal additive manufacturing. *Computational Materials Science*, 141 (2018), pp. 207-220.

SNE Simulation News

EUROSIM Data and Quick Info

DBSS
Dutch Benelux
Simulation Society

VESS – Virtual EUROSIM Seminar

Virtual Simulation Presentations, from June 2020 on www.eurosim2023.eu

SIMS
Scandinavian
Simulation Society

SIMS EUROSIM Conference 2021

Sept. 22-24, 2021, Oulu, Finland www.scansims.org

MATHMOD

Vienna 2022

Feb. 16-18, 2022, Vienna, Austria www.mathmod.at

DBSS
Dutch Benelux
Simulation Society

EUROSIM CONGRESS 2023

Spring/Autumn 2023, Amsterdam, The Netherlands www.eurosim2023.eu

Contents

Short Info EUROSIM	N2
Short Info ASIM, CEA-SMSG	N3
Short Info CSSS, DBSS, LIOPHANT, LSS	N4
Short Info KA-SIM, MIMOS, NSSM, PSCS	N5
Short Info SIMS, SLOSIM, UKSIM	N6
Short Info ROMSIM, Albanian Society	N7
Short Info ARGESIM, SNE	N8
VESS – Virtual EUROSIM Simulation Seminar Series and EUROSIM Conferences & Congress	Cover Inside

Simulation Notes Europe SNE is the official membership journal of EUROSIM and distributed / available to members of the EUROSIM Societies as part of the membership benefits.

If you have any information, announcement, etc. you want to see published, please contact a member of the editorial board in your country or the editorial office. For scientific publications, please contact the EiC.

This *EUROSIM Data & Quick Info* compiles data from EUROSIM societies and groups: addresses, weblinks, and officers of societies with function and email, to be published regularly in SNE issues. This information is also published at EUROSIM's website www.eurosim.info.

SNE Reports Editorial Board

EUROSIM Miguel Mujica Mota, m.mujica.mota@hva.nl
 Nikolas Popper, niki.popper@dwht.at
 ASIM A. Körner, andreas.koerner@tuwien.ac.at
 CEA-SMSG Emilio Jiménez, emilio.jimenez@unirioja.es
 CSSS Mikuláš Alexík, alexik@frtk.utc.sk
 DBSS M. Mujica Mota, m.mujica.mota@hva.nl
 LIOPHANT F. Longo, f.longo@unical.it
 LSS Juri Tolujew, juri.tolujew@iff.fraunhofer.de
 KA-SIM Edmond Hajrizi, info@ka-sim.com
 MIMOS Paolo Proietti, roma@mimos.it
 NSSM Y. Senichenkov, senyb@dcn.icc.spbstu.ru
 PSCS Zenon Sosnowski, zenon@ii.pb.bialystok.pl
 SIMS Esko Juuso, esko.juuso@oulu.fi
 SLOSIM Vito Logar, vito.logar@fe.uni-lj.si
 UKSIM David Al-Dabass, david.al-dabass@ntu.ac.uk
 ROMSIM Constanta Zoe Radulescu, zoe@ici.ro
 ALBSIM Majlinda Godolja, majlinda.godolja@feut.edu.al

SNE Editorial Office /ARGESIM

→ www.sne-journal.org, www.eurosim.info

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

✉ SNE Editorial Office

Johannes Tänzler (Layout, Organisation)
 Irmgard Husinsky (Web, Electronic Publishing)
 Felix Breitenacker EiC (Organisation, Authors)
 ARGESIM/Math. Modelling & Simulation Group,
 Inst. of Analysis and Scientific Computing, TU Wien
 Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria



EUROSIM Federation of European Simulation Societies

General Information. EUROSIM, the Federation of European Simulation Societies, was set up in 1989. The purpose of EUROSIM is to provide a European forum for simulation societies and groups to promote modelling and simulation in industry, research, and development – by publication and conferences. → www.eurosim.info

Member Societies. EUROSIM members may be national simulation societies and regional or international societies and groups dealing with modelling and simulation. At present EUROSIM has *Full Members* and *Observer Members* (*), and *Member Candidates* (**).

ASIM	Arbeitsgemeinschaft Simulation <i>Austria, Germany, Switzerland</i>
CEA-SMSG	Spanish Modelling and Simulation Group; <i>Spain</i>
CSSS	Czech and Slovak Simulation Society <i>Czech Republic, Slovak Republic</i>
DBSS	Dutch Benelux Simulation Society <i>Belgium, Netherlands</i>
KA-SIM	Kosovo Simulation Society, <i>Kosovo</i>
LIOPHANT	LIOPHANT Simulation Club; <i>Italy & International</i>
LSS	Latvian Simulation Society; <i>Latvia</i>
PSCS	Polish Society for Computer Simulation; <i>Poland</i>
MIMOS	Italian Modelling and Simulation Ass.; <i>Italy</i>
NSSM	Russian National Simulation Society <i>Russian Federation</i>
SIMS	Simulation Society of Scandinavia <i>Denmark, Finland, Norway, Sweden</i>
SLOSIM	Slovenian Simulation Society; <i>Slovenia</i>
UKSIM	United Kingdom Simulation Society <i>UK, Ireland</i>
ALBSIM	Albanian Simulation Society*; <i>Albania</i>
ROMSIM	Romanian Society for Modelling and Simulation*; <i>Romania</i>
Societies in Re-Organisation:	
CROSSIM	<i>Croatian Society f. Simulation Modeling; Croatia</i>
FRANCO-SIM	<i>Société Francophone de Simulation Belgium, France</i>
HSS	<i>Hungarian Simulation Society; Hungary</i>
ISCS	<i>Italian Society for Computer Simulation, Italy</i>

EUROSIM Board / Officers. EUROSIM is governed by a board consisting of one representative of each member society, and president, past president, and SNE representative. The President is nominated by the society organising the next EUROSIM Congress. Secretary, and Treasurer are elected out of members of the board.

President	M. Mujica Mota (DBSS), <i>m.mujica.mota@hva.nl</i>
Past President	Emilio Jiménez (CAE-SMSG), <i>emilio.jimenez@unirioja.es</i>
Secretary	Niki Popper, <i>niki.popper@dwh.at</i>
Treasurer	Felix Breitenecker (ASIM) <i>felix.breitenecker@tuwien.ac.at</i>
Webmaster	Irmgard Husinsky, <i>irmgard.husinsky@tuwien.ac.at</i>
SNE Editor	F. Breitenecker, <i>eic@sne-journal.org</i>

SNE – Simulation Notes Europe. SNE is EUROSIM's scientific journal with peer reviewed contributions as well as a membership journal for EUROSIM with information from the societies. EUROSIM societies distribute SNE (electronic or printed) to their members as official membership journal. SNE Publishers are EUROSIM, ARGESIM and ASIM.

SNE	Felix Breitenecker
Editor-in-Chief	<i>eic@sne-journal.org</i>

→ www.sne-journal.org, ☎ office@sne-journal.org

EUROSIM Congress and Conferences.

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

EUROSIM Congress 2019, the 10th EUROSIM Congress, was organised by CEA-SMSG, the Spanish Simulation Society, in La Rioja, Logroño, Spain, July 1-5, 2019; → www.eurosim2019.com

Due to Covid-19 virus in 2020 no EUROSIM events take place. To bridge this gap, EUROSIM is organising the series VESS - Virtual EUROSIM Simulation Seminar – seminars by simulation professionals (2 hours via web), in preparation for upcoming EUROSIM events. → www.eurosim2023.eu

Next main event is SIMS EUROSIM Conference 2021, September 21–23, 2021, Oulu, Finland. SIMS, the Scandinavian simulation society, extends every third year the annual SIMS Conference to the SIMS EUROSIM Conference. → www.scansims.org

MATHMOD Vienna. This triennial EUROSIM Conference is mainly organized by ASIM, the German simulation society, and ARGESIM, with main co-sponsor IFAC.

MATHMOD 2022, the 10th MATHMOD Vienna Conference on Mathematical Modelling will take place in Vienna, February 16-18, 2022. → www.mathmod.at

EUROSIM Congress 2023, the 11th EUROSIM Congress, will be organised by DBSS, the Dutch Benelux simulation society, in Amsterdam, Spring/Autumn 2023. → www.eurosim2023.eu

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

→ www.liophant.org/conferences/

EUROSIM Member Societies



ASIM
German Simulation Society
Arbeitsgemeinschaft Simulation

ASIM (Arbeitsgemeinschaft Simulation) is the association for simulation in the German speaking area, servicing mainly Germany, Switzerland and Austria. ASIM was founded in 1981 and has now about 400 individual members (including associated), and 90 institutional or industrial members.

→ www.asim-gi.org with members' area

✉ info@asim-gi.org, admin@asim-gi.org

✉ ASIM – Inst. of Analysis and Scientific Computing
Vienna University of Technology (TU Wien)
Wiedner Hauptstraße 8-10, 1040 Vienna, Austria

ASIM Officers

President	Felix Breiteneker felix.breiteneker@tuwien.ac.at
Vice presidents	Sigrid Wenzel, s.wenzel@uni-kassel.de T. Pawletta, thorsten.pawletta@hs-wismar.de A. Körner, andreas.koerner@tuwien.ac.at
Secretary	Ch. Deatcu, christina.deatcu@hs-wismar.de I. Husinsky, Irmgard.husinsky@tuwien.ac.at
Membership Affairs	S. Wenzel, s.wenzel@uni-kassel.de Ch. Deatcu, christina.deatcu@hs-wismar.de F. Breiteneker, felix.breiteneker@tuwien.ac.at
Repr. EUROSIM	F. Breiteneker, felix.breiteneker@tuwien.ac.at A. Körner, andreas.koerner@tuwien.ac.at
Internat. Affairs – GI Contact	O. Rose, Oliver.Rose@tu-dresden.de N. Popper, niki.popper@dwh.at
Editorial Board SNE	T. Pawletta, thorsten.pawletta@hs-wismar.de Ch. Deatcu, christina.deatcu@hs-wismar.de
Web EUROSIM	I. Husinsky, Irmgard.husinsky@tuwien.ac.at

Last data update April 2020

ASIM is organising / co-organising the following international conferences:

- ASIM Int. Conference 'Simulation in Production and Logistics' – biannual
- ASIM 'Symposium Simulation Technique' – biannual
- MATHMOD Int. Vienna Conference on Mathematical Modelling – triennial

Furthermore, ASIM is co-sponsor of WSC - Winter Simulation Conference, of SCS conferences *SpringSim* and *SummerSim*, and of *I3M* and *Simutech* conference series.

ASIM Working Committees

GMMS	Methods in Modelling and Simulation Th. Pawletta, thorsten.pawletta@hs-wismar.de
SUG	Simulation in Environmental Systems Jochen Wittmann, wittmann@informatik.uni-hamburg.de
STS	Simulation of Technical Systems Walter Commerell, commerell@hs-ulm.de
SPL	Simulation in Production and Logistics Sigrid Wenzel, s.wenzel@uni-kassel.de
Edu	Simulation in Education/Education in Simulation A. Körner, andreas.koerner@tuwien.ac.at
BIG DATA	Working Group Data-driven Simulation in Life Sciences; niki.popper@dwh.at
WORKING GROUPS	Simulation in Business Administration, in Traffic Systems, for Standardisation, etc.

CEA-SMSG – Spanish Modelling and Simulation Group

CEA is the Spanish Society on Automation and Control and it is the national member of IFAC (International Federation of Automatic Control) in Spain. Since 1968 CEA-IFAC looks after the development of the Automation in Spain, in its different issues: automatic control, robotics, *SIMULATION*, etc. The association is divided into national thematic groups, one of which is centered on Modeling, Simulation and Optimization, constituting the CEA Spanish Modeling and Simulation Group (CEA-SMSG). It looks after the development of the Modelling and Simulation (M&S) in Spain, working basically on all the issues concerning the use of M&S techniques as essential engineering tools for decision-making and optimization.

→ <http://www.ceautomatica.es/grupos/>

→ emilio.jimenez@unirioja.es

simulacion@cea-ifac.es

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

CEA - SMSG Officers

President	Emilio Jiménez, emilio.jimenez@unirioja.es
Vice president	Juan Ignacio Latorre, juanignacio.latorre@unavarra.es
Repr. EUROSIM	Emilio Jiménez, emilio.jimenez@unirioja.es
Edit. Board SNE	Juan Ignacio Latorre, juanignacio.latorre@unavarra.es
Web EUROSIM	Mercedes Perez mercedes.perez@unirioja.es

Last data update February 2018



CSSS – Czech and Slovak Simulation Society

CSSS -The *Czech and Slovak Simulation Society* has about 150 members working in Czech and Slovak national scientific and technical societies (*Czech Society for Applied Cybernetics and Informatics*, *Slovak Society for Applied Cybernetics and Informatics*). CSSS main objectives are: development of education and training in the field of modelling and simulation, organising professional workshops and conferences, disseminating information about modelling and simulation activities in Europe. Since 1992, CSSS is full member of EUROSIM.

→ www.fit.vutbr.cz/CSSS

✉ snorek@fel.cvut.cz

- ✉ CSSS / Miroslav Šnorek, CTU Prague
FEE, Dept. Computer Science and Engineering,
Karlovo nám. 13, 121 35 Praha 2, Czech Republic

CSSS Officers

President	Miroslav Šnorek, snorek@fel.cvut.cz
Vice president	Mikuláš Alexik, alexik@frtk.fri.utc.sk
Scientific Secr.	A. Kavička, Antonin.Kavicka@upce.cz
Repr. EUROSIM	Miroslav Šnorek, snorek@fel.cvut.cz
Edit. Board SNE	Mikuláš Alexik, alexik@frtk.fri.utc.sk
Web EUROSIM	Petr Peringer, peringer@fit.vutbr.cz

Last data update December 2012

DBSS – Dutch Benelux Simulation Society

The *Dutch Benelux Simulation Society* (DBSS) was founded in July 1986 in order to create an organisation of simulation professionals within the Dutch language area. DBSS has actively promoted creation of similar organisations in other language areas. DBSS is a member of EUROSIM and works in close cooperation with its members and with affiliated societies.

→ www.DutchBSS.org

✉ a.w.heemink@its.tudelft.nl

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

DBSS Officers

President	M. Mujica Mota, m.mujica.mota@hva.nl
Vice president	A. Heemink, a.w.heemink@its.tudelft.nl
Treasurer	A. Heemink, a.w.heemink@its.tudelft.nl
Secretary	P. M. Scala, p.m.scala@hva.nl
Repr. EUROSIM	M. Mujica Mota, m.mujica.mota@hva.nl
Edit. SNE/Web	M. Mujica Mota, m.mujica.mota@hva.nl

Last data update June 2016



LIOPHANT Simulation

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

→ www.liophant.org

✉ info@liophant.org

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

LIOPHANT Officers

President	A.G. Bruzzone, agostino@itim.unige.it
Director	E. Bocca, enrico.bocca@liophant.org
Secretary	A. Devoti, devoti.a@iveco.com
Treasurer	Marina Massei, massei@itim.unige.it
Repr. EUROSIM	A.G. Bruzzone, agostino@itim.unige.it
Deputy	F. Longo, f.longo@unical.it
Edit. Board SNE	F. Longo, f.longo@unical.it
Web EUROSIM	F. Longo, f.longo@unical.it

Last data update June 2016

LSS – Latvian Simulation Society

The Latvian Simulation Society (LSS) has been founded in 1990 as the first professional simulation organisation in the field of Modelling and simulation in the post-Soviet area. Its members represent the main simulation centres in Latvia, including both academic and industrial sectors.

→ www.itl.rtu.lv/imb/

✉ merkur@itl.rtu.lv

- ✉ LSS / Yuri Merkuryev, Dept. of Modelling
and Simulation Riga Technical University
Kalku street 1, Riga, LV-1658, LATVIA

LSS Officers

President	Yuri Merkuryev, merkur@itl.rtu.lv
Vice President	Egils Ginters, egils.ginters@rtu.lv
Secretary	Artis Teilans, artis.teilans@rta.lv
Repr. EUROSIM	Egils Ginters, egils.ginters@rtu.lv
Deputy	Artis Teilans, artis.teilans@rta.lv
Edit. Board SNE	Juri Tolujew, Juri.Tolujew@iff.fraunhofer.de
Web EUROSIM	Vitaly Bolshakov, vitalijs.bolsakovs@rtu.lv

Last data update June 2019

KA-SIM Kosovo Simulation Society

Kosova Association for Modeling and Simulation (KA-SIM, founded in 2009), is part of Kosova Association of Control, Automation and Systems Engineering (KA-CASE). KA-CASE was registered in 2006 as non Profit Organization and since 2009 is National Member of IFAC – International Federation of Automatic Control. KA-SIM joined EUROSIM as Observer Member in 2011. In 2016, KA-SIM became full member.

KA-SIM has about 50 members, and is organizing the international conference series International Conference in Business, Technology and Innovation, in November, in Durrhës, Albania, and IFAC Simulation Workshops in Pristina.

→ www.ubt-uni.net/ka-case

✉ ehajrizi@ubt-uni.net

✉ MOD&SIM KA-CASE; Att. Dr. Edmond Hajrizi
Univ. for Business and Technology (UBT)
Lagjja Kalabria p.n., 10000 Prishtina, Kosovo

KA-SIM Officers

President	Edmond Hajrizi, ehajrizi@ubt-uni.net
Vice president	Muzafer Shala, info@ka-sim.com
Secretary	Lulzim Beqiri, info@ka-sim.com
Treasurer	Selman Berisha, info@ka-sim.com
Repr. EUROSIM	Edmond Hajrizi, ehajrizi@ubt-uni.net
Deputy	Muzafer Shala, info@ka-sim.com
Edit. Board SNE	Edmond Hajrizi, ehajrizi@ubt-uni.net
Web EUROSIM	Betim Gashi, info@ka-sim.com

Last data update December 2016

MIMOS – Italian Modelling and Simulation Association

MIMOS (Movimento Italiano Modellazione e Simulazione – Italian Modelling and Simulation Association) is the Italian association grouping companies, professionals, universities, and research institutions working in the field of modelling, simulation, virtual reality and 3D, with the aim of enhancing the culture of ‘virtuality’ in Italy, in every application area.

MIMOS became EUROSIM Observer Member in 2016 and EUROSIM Full Member in September 2018.

→ www.mimos.it

✉ roma@mimos.it – info@mimos.it

✉ MIMOS – Movimento Italiano Modellazione e Simulazione; via Ugo Foscolo 4, 10126 Torino – via Laurentina 760, 00143 Roma

MIMOS Officers

President	Paolo Proietti, roma@mimos.it
Secretary	Davide Borra, segreteria@mimos.it
Treasurer	Davide Borra, segreteria@mimos.it
Repr. EUROSIM	Paolo Proietti, roma@mimos.it
Deputy	Agostino Bruzzone, agostino@itim.unige.it
Edit. Board SNE	Paolo Proietti, roma@mimos.it

Last data update December 2016

NSSM – National Society for Simulation Modelling (Russia)

NSSM - The Russian National Simulation Society (Национальное Общество Имитационного Моделирования – НОИМ) was officially registered in Russian Federation on February 11, 2011. In February 2012 NSS has been accepted as an observer member of EUROSIM, and in 2015 NSSM has become full member.

→ www.simulation.su

✉ yusupov@ias.spb.su

✉ NSSM / R. M. Yusupov,
St. Petersburg Institute of Informatics and Automation
RAS, 199178, St. Petersburg, 14th lin. V.O, 39

NSSM Officers

President	R. M. Yusupov, yusupov@ias.spb.su
Chair Man. Board	A. Plotnikov, plotnikov@sstc.spb.ru
Secretary	M. Dolmatov, dolmatov@simulation.su
Repr. EUROSIM	R.M. Yusupov, yusupov@ias.spb.su Y. Senichenkov, senyb@dcn.icc.spbstu.ru
Deputy	B. Sokolov, sokol@ias.spb.su
Edit. Board SNE	Y. Senichenkov, senyb@mail.ru , senyb@dcn.icc.spbstu.ru

Last data update February 2018

PSCS – Polish Society for Computer Simulation

PSCS was founded in 1993 in Warsaw. PSCS is a scientific, non-profit association of members from universities, research institutes and industry in Poland with common interests in variety of methods of computer simulations and its applications. At present PSCS counts 257 members.



→ www.eurosim.info, www.ptsk.pl/

✉ leon@ibib.waw.pl

✉ PSCS / Leon Bobrowski, c/o IBIB PAN,
ul. Trojdena 4 (p.416), 02-109 Warszawa, Poland

PSCS Officers	
President	Leon Bobrowski, leon@ibib.waw.pl
Vice president	Tadeusz Nowicki, Tadeusz.Nowicki@wat.edu.pl
Treasurer	Z. Sosnowski, zenon@ii.pb.bialystok.pl
Secretary	Zdzisław Galkowski, Zdzislaw.Galkowski@simr.pw.edu.pl
Repr. EUROSIM	Leon Bobrowski, leon@ibib.waw.pl
Deputy	Tadeusz Nowicki, tadeusz.nowicki@wat.edu.pl
Edit. Board SNE	Zenon Sosnowski, z.sosnowski@pb.edu.pl
Web EuroSim	Magdalena Topczewska m.topczewska@pb.edu.pl

Last data update December 2013

SIMS – Scandinavian Simulation Society

SIMS is the *Scandinavian Simulation Society* with members from the five Nordic countries Denmark, Finland, Iceland, Norway and Sweden. The SIMS history goes back to 1959. SIMS practical matters are taken care of by the SIMS board consisting of two representatives from each Nordic country (Iceland one board member).

SIMS Structure. SIMS is organised as federation of regional societies. There are **FinSim** (Finnish Simulation Forum), **MoSis** (Society for Modelling and Simulation in Sweden), **DKSIM** (Dansk Simuleringsforening) and **NFA** (Norsk Forening for Automatisering).

→ www.scansims.org

✉ bernt.lie@usn.no

✉ SIMS / Bernt Lie, Faculty of Technology, Univ.College of Southeast Norway, Department of Technology, Kjølens ring 56, 3914 Porsgrunn, Norway

SIMS Officers	
President	Bernt Lie, Bernt.Lie@usn.no
Vice president	Erik Dahlquist, erik.dahlquist@mdh.se
Treasurer	Vadim Engelson, vadime@mathcore.com
Repr. EUROSIM	Esko Juuso, esko.juuso@oulu.fi
Edit. Board SNE	Esko Juuso, esko.juuso@oulu.fi
Web EuroSim	Vadim Engelson, vadime@mathcore.com

Last data update February 2020



SLOSIM – Slovenian Society for Simulation and Modelling

SLOSIM - Slovenian Society for Simulation and Modelling was established in 1994 and became the full member of EUROSIM in 1996. Currently it has 90 members from both Slovenian universities, institutes, and industry. It promotes modelling and simulation approaches to problem solving in industrial as well as in academic environments by establishing communication and cooperation among corresponding teams.

→ www.slosim.si

✉ slosim@fe.uni-lj.si

✉ SLOSIM / Vito Logar, Faculty of Electrical Engineering, University of Ljubljana,
Tržaška 25, 1000 Ljubljana, Slovenia

SLOSIM Officers	
President	Vito Logar, vito.logar@fe.uni-lj.si
Vice president	Božidar Šarler, bozidar.sarler@ung.si
Secretary	Simon Tomažič, simon.tomazic@fe.uni-lj.si
Treasurer	Milan Simčič, milan.simcic@fe.uni-lj.si
Repr. EUROSIM	B. Zupančič, borut.zupancic@fe.uni-lj.si
Deputy	Vito Logar, vito.logar@fe.uni-lj.si
Edit. Board SNE	R. Karba, rihard.karba@fe.uni-lj.si
Web EuroSim	Vito Logar, vito.logar@fe.uni-lj.si

Last data update December 2018

UKSIM - United Kingdom Simulation Society

The UK Simulation Society is very active in organizing conferences, meetings and workshops. UKSim holds its annual conference in the March-April period. In recent years the conference has always been held at Emmanuel College, Cambridge. The Asia Modelling and Simulation Section (AMSS) of UKSim holds 4-5 conferences per year including the EMS (European Modelling Symposium), an event mainly aimed at young researchers, organized each year by UKSim in different European cities. Membership of the UK Simulation Society is free to participants of any of our conferences and their co-authors.

→ uksim.info

✉ david.al-dabass@ntu.ac.uk

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

UKSIM Officers	
President	David Al-Dabass, <i>david.al-dabass@ntu.ac.uk</i>
Secretary	T. Bashford, <i>tim.bashford@uwtsd.ac.uk</i>
Treasurer	D. Al-Dabass, <i>david.al-dabass@ntu.ac.uk</i>
Membership chair	G. Jenkins, <i>glenn.l.jenkins@smu.ac.uk</i>
Local/Venue chair	Richard Cant, <i>richard.cant@ntu.ac.uk</i>
Repr. EUROSIM	Dr Taha Osman, <i>taha.osman@ntu.ac.uk</i>
Deputy	T. Bashford, <i>tim.bashford@uwtsd.ac.uk</i>
Edit. Board SNE	D. Al-Dabass, <i>david.al-dabass@ntu.ac.uk</i>

Last data update March 2020

EUROSIM Observer Members

ROMSIM – Romanian Modelling and Simulation Society

ROMSIM has been founded in 1990 as a non-profit society, devoted to theoretical and applied aspects of modelling and simulation of systems. ROMSIM currently has about 100 members from Romania and Moldavia.

→ www.eurosim.info/societies/romsim/

✉ florin_h2004@yahoo.com

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

ROMSIM Officers	
President	N. N.
Vice president	Florin Hartescu, <i>florin_h2004@yahoo.com</i> Marius Radulescu, <i>mrادulescu.csmro@yahoo.com</i>
Repr. EUROSIM	Marius Radulescu, <i>mrادulescu.csmro@yahoo.com</i>
Deputy	Florin Hartescu, <i>florin_h2004@yahoo.com</i>
Edit. Board SNE	Constanta Zoe Radulescu, <i>zoe@ici.ro</i>
Web EUROSIM	Florin Hartescu, <i>florin_h2004@yahoo.com</i>

Last data update June 2019

ALBSIM – Albanian Simulation Society

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

The society is involved in different international and local simulation projects, and is engaged in the organisation of the conference series ISTI - Information Systems and Technology. In July 2019 the society was accepted as EUROSIM Observer Member.

→ www.eurosim.info/societies/albsim/

✉ kozeta.sevrani@unitir.edu.al

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

Albanian Simulation Society- Officers	
Chairt	Kozeta Sevrani, <i>kozeta.sevrani@unitir.edu.al</i>
Repr. EUROSIM	Kozeta Sevrani, <i>kozeta.sevrani@unitir.edu.al</i>
Edit. Board SNE	Albana Gorishti, <i>albana.gorishti@unitir.edu.al</i> Majlinda Godolja, <i>majlinda.godolja@feut.edu.al</i>

Last data update July 2019

Societies in Re-Organisation

The following societies are at present inactive or under re-organisation:

- CROSSIM - Croatian Society for Simulation Modelling
- FRANCO-SIM – Société Francophone de Simulation
- HSS – Hungarian Simulation Society
- ISCS – Italian Society for Computer Simulation



Association Simulation News



ARGESIM is a non-profit association generally aiming for dissemination of information on system simulation – from research via development to applications of system simulation. ARGESIM is closely co-operating with EUROSIM, 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).

→ www.argesim.org

✉ → office@argesim.org

✉ → ARGESIM/Math. Modelling & Simulation Group,
Inst. of Analysis and Scientific Computing, TU Wien
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria
Attn. Prof. Dr. Felix Breiteneker

ARGESIM is following its aims and scope by the following activities and projects:

- 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*
- Publication of the series ARGESIM Reports for monographs in system simulation, and proceedings of simulation conferences and workshops
- 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
- Organisation of Seminars and Summerschools on Simulation
- Administration of ASIM (German Simulation Society) and administrative support for EUROSIM www.eurosim.info
- Support of ERASMUS and CEEPUS activities in system simulation for TU Wien

ARGESIM is a registered non-profit association and a registered publisher: ARGESIM Publisher Vienna, root ISBN 978-3-901608-xx-y, root DOI 10.11128/z...zz.zz. Publication is open for ASIM and for EUROSIM Member Societies.

SNE – Simulation Notes Europe

SNE

The scientific journal *SNE – Simulation Notes Europe* 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 puts special emphasis on the overall view in simulation, and on comparative investigations. Furthermore, SNE welcomes contributions on education in/for/with simulation.

SNE is also the forum for the ARGESIM Benchmarks on *Modelling Approaches and Simulation Implementations* publishing benchmarks definitions, solutions, reports and studies – including model sources via web.

→ www.sne-journal.org,

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

✉ → SNE Editorial Office

ARGESIM/Math. Modelling & Simulation Group,
Inst. of Analysis and Scientific Computing, TU Wien
Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria
Eic Prof. Dr. Felix Breiteneker

SNE, primarily an electronic journal, follows an open access strategy, with free download in basic layout. SNE is the official membership journal of EUROSIM, the *Federation of European Simulation Societies*. Members of EUROSIM Societies are entitled to download SNE in high-quality, and to access additional sources of benchmark publications, model sources, etc. On the other hand, SNE offers EUROSIM Societies a publication forum for post-conference publication of the society's international conferences, and the possibility to compile thematic or event-based SNE Special Issues.

Simulationists are invited to submit contributions of any type – *Technical Note*, *Short Note*, *Project Note*, *Educational Note*, *Benchmark Note*, etc. via SNE's website:

SNE

SIMULATION NOTES EUROPE

Official Membership Journal of EUROSIM

Home	Aims and Scope	Editorial Board	SNE Volumes	Search
----------------------	--------------------------------	---------------------------------	-----------------------------	------------------------

Submission
Submission Confirmation

[SNE > Contribute / Contact > Submission](#)

Manuscript Submission

Article Title: *

Type: Technical Note

Corresponding Author: *

E-Mail: *

Upload your Article (PDF): *



ASIM



ASIM



ASIM



ASIM Books – ASIM Book Series – ASIM Buchreihen

Proceedings

Simulation in Production and Logistics 2019 – 18. ASIM Fachtagung Simulation in Produktion und Logistik
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; 10.11128/arep.56; ARGESIM Report 56; ASIM Mitteilung AM 168

Simulation in Production and Logistics 2017 – 17. ASIM Fachtagung Simulation in Produktion und Logistik
Sigrid Wenzel, Tim Peter (Hrsg.); ISBN Print 978-3-7376-0192-4, ISBN Online 978-3-7376-0193-1, Kassel university
press GmbH, Kassel, 2017; ASIM Mitteilung AM164

Tagungsband ASIM SST 2016 - 23. Symposium Simulationstechnik, HTW Dresden, September 2016
T. Wiedemann (Hrsg.); ARGESIM Verlag Wien, 2016; ISBN ebook 978-3-901608-49-0;
ARGESIM Report 52; ASIM Mitteilung AM 160

Books

Kostensimulation - Grundlagen, Forschungsansätze, Anwendungsbeispiele

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

Simulation und Optimierung in Produktion und Logistik – Praxisorientierter Leitfaden mit Fallbeispielen.

L. März, W. Krug, O. Rose, G. Weigert (Hrsg.); ISBN 978-3-642-14535-3, Springer, 2011; AM 130

Book Series Fortschrittsberichte Simulation – Advances in Simulation

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; ISBN print 978-3-903311-07-7, TUVerlag Wien, 2019

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: FBS19;
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

Parlez-vous MATLAB?

Über eine Million Menschen weltweit sprechen MATLAB. Ingenieure und Wissenschaftler in allen Bereichen – von der Luft- und Raumfahrt über die Halbleiterindustrie bis zur Biotechnologie, Finanzdienstleistungen und Geo- und Meereswissenschaften – nutzen MATLAB, um ihre Ideen auszudrücken. Sprechen Sie MATLAB?



Modellierung eines elektrischen Potentials in einem Quantum Dot.

Dieses Beispiel finden Sie unter:
www.mathworks.de/te

MATLAB®
The language of technical computing

Image: Kim Young-Sang, Jang-Ho Kim, Quantum Device Lab, HanYang Univ (©2011 The MathWorks, Inc.)

