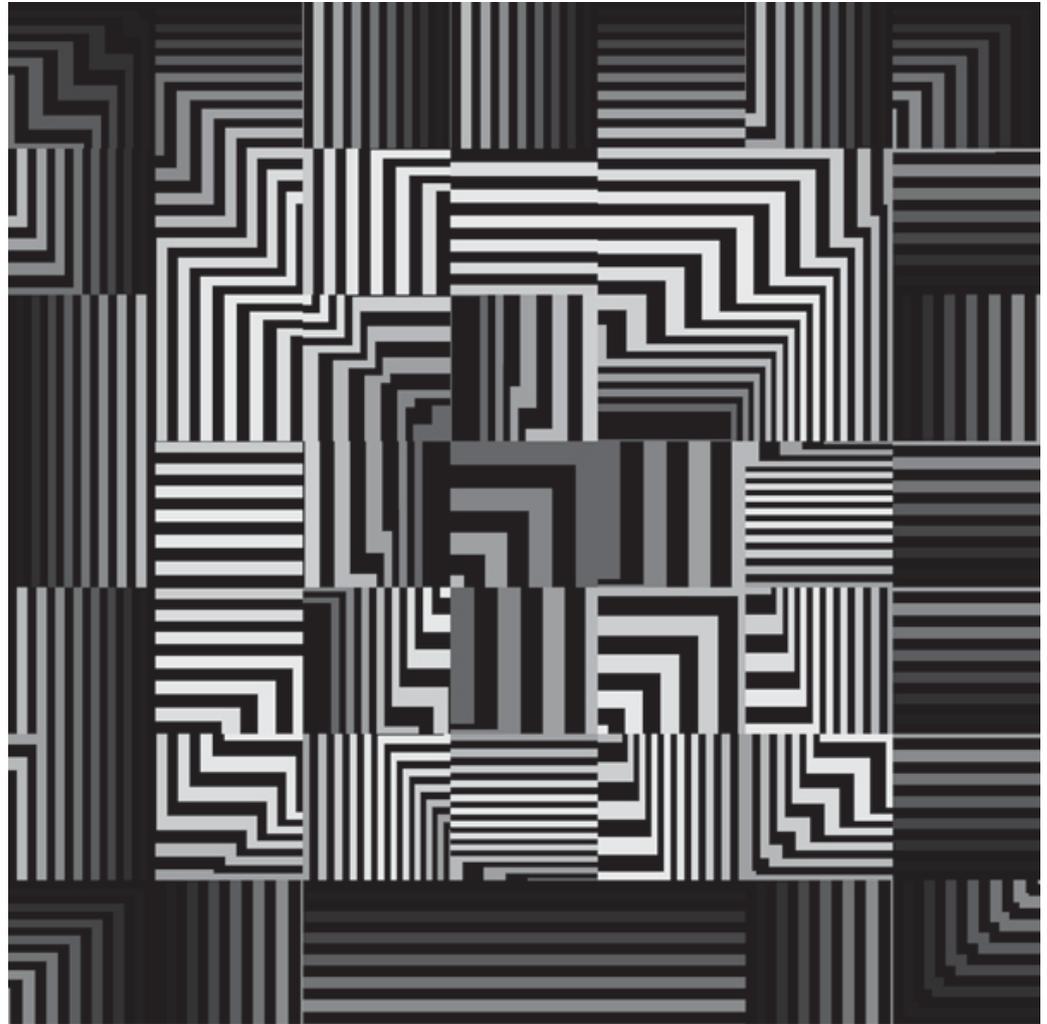


SNE SIMULATION NOTES EUROPE



Journal on Developments and Trends in Modelling and Simulation

EUROSIM Scientific Membership Journal

Vol. 30 No.2, June 2020

ISSN Online 2306-0271

DOI 10.11128/sne.30.2.1051

ISSN Print 2305-9974

ISBN Print 978-3-903311-06-0

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.



The banner features the EUROSIM logo on the left, the DBSS logo (Dutch Benelux Simulation Society) in the center, and the text "VESS - VIRTUAL EUROSIM SIMULATION SEMINAR free online seminar series" on the right. The website address "www.eurosim2023.eu" is displayed at the bottom left.

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 banner shows a conference room with tables and chairs. The text reads "SIMS EUROSIM 2021 Conference on Modelling and Simulation 21-23 September 2021 in Oulu, Finland".

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



The banner features the MATHMOD 2022 logo on the left, the text "10th Vienna International Conference on Mathematical Modelling" in the center, and the dates "February 16 - 18, 2022, Vienna, Austria" on the right. The background shows a building facade.

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



The banner features the text "EUROSIM 2023" in large yellow letters on the left, "Amsterdam Spring/Autumn 2023" in the center, and the DBSS logo (Dutch Benelux Simulation Society) on the right. The background shows a city street at night.

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

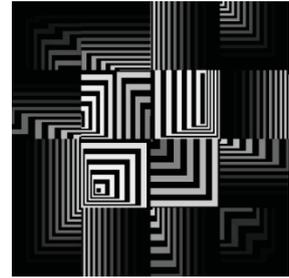
Dear Readers, SNE in times of Covid-19 virus – a new challenge for us. Due to the cancellation of conferences and workshops we had to re-schedule our publication plans, especially as foreseen post-conference publication for this issue SNE 30(2) from spring 2020 conferences are not available. We asked authors to submit announced publications earlier, we encouraged the review board to work with shorter deadlines, and so we are happy to finish electronic SNE 30(2), June 2020, in time. We are proud on two novelties – ‘high math meets simulation’ and ‘students present simulation projects’. In the first publication, M. Holzinger indeed lets meet higher math with simulation: an introduction to PDE solution by conformal mappings, to be continued by high math extension and a surprising application in the next SNE issue.

On the other side, we start in this issue with a new category of Notes – Student Notes: ‘simulation teachers’ may invite their good students to report on a finished student simulation project – in style and presentation different from a classical scientific paper (and with different review) – in this issue forest growth simulation with harvesting, draught, and bark beetle. Thanks to the reviewers we could publish the contributions by Hai Nguyen Van et al. on aviation scenarios and by O. Ullrich et al. on repositories for simulation data. And last but not least we had two ‘last’ post-conference publication in line, modelling for gluing of particle boards by C Rößler and M. Riegler, and a Short Note on Co-Simulation by G. Schweiger et al. That makes six publications for SNE 30(2).

Thirty Years SNE - we are pleased that for SNE Volume 30 again Vlatko Čerić, past president of the Croatian Simulation Society, provides his algorithmic art as design for SNE cover pages. The artist and simulationist Vlatko Čerić has chosen four algorithmic art pictures from the series LABYRINTH for covers of SNE Volume 30. The cover of this issue SNE 30(2) presents LABYRINTH no. 7, and at right and a cover preview for the cover for SNE 30(3) - LABYRINTH no. 17. For further info, please visit vceric.net.

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

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



Contents SNE 30(2)

Online SNE 30(2), DOI 10.11128/sne.30.2.1051
ARGESIM Publisher, Vienna, www.argesim.org

Print SNE 30(2) ISBN 978-3-903311-06-0
TU Verlag Vienna, Print-on-Demand, www.tuverlag.at

CTDS – Mauling Heat Equation on Unit Disk by Conformal Parametrization. <i>M. Holzinger</i>	43
Timed Discrete-Event Simulation of Aviation Scenarios. <i>H. Nguyen Van, F. Boulanger, B. Wolff</i>	51
Examining Repositories for Simulation Data. <i>O. Ullrich, V. Potapenko, N. Rishe</i>	61
Simulation in Wood Science: a Novel Model for the Process Gluing. <i>C. Rößler, M. Riegler</i>	67
Co-Simulation - An Empirical Survey: Applications, Recent Developments and Future Challenges. <i>G. Schweiger, G. Engel, J.-P. Schöggel, I. Hafner, T. S. Noudui, C. Gomes</i>	73
MATLAB-based Modelling and Simulation of Forest Development: Growth – Harvesting – Draught – Bark Beetle – Biomass Plant Usage. <i>C. Königswieser, A. Jahn</i>	77
EUROSIM Societies Short Info	N1 – N8

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.eurosim.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
Universita 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 Merkurjev, 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

CTDS – Mauling Heat Equation on Unit Disk by Conformal Parametrization

Martin Holzinger

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

SNE 30(2), 2020, 43-50, DOI: 10.11128/sne.30.tn.10511
Received: March 15, 2020; Revised April 24, 2020
Accepted: April 30, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Let us consider a heat conduction problem on the unit square, solve it analytically and compare this series solution with both the results obtained by MATLAB/PDE-Toolbox (using the FE-method) and a self-implementation achieved with *Mathematica* (using the method of lines). Extending Finite-Difference formulae to higher precision gives rise to the idea of utilizing the CTDS-method, best suitable on regular and equidistant grids, also on other domains. By introducing apt coordinates one is therefore able to do a parametrization, e.g. of the unit disk, by the square. Conformal transformation from square to disk provides this parametrization, the original implementation can easily be extended and we find by again comparing a series solution to our obtained simulation results, that order of convergence is being preserved. Moreover, our conformal transformation provides the fundamental tensor and no further structural errors are being introduced as the involved elliptic functions can be evaluated to arbitrary precision.

Introduction

By application of the CTDS-method ("Continuous Time Discrete Space") spatial derivatives in a PDE are replaced by their Finite-Difference approximations which yields a coupled system of ODEs getting stiffer with grid refinement. These systems can be treated with standard algorithms and we show that order of convergence can easily be adjusted to the requirements. Restricted to the unit square and regular grids, conformal maps then provide access to more general domains.

1 Series Solution on Unit Square

For a given heat conduction problem on unit square Ω :

$$u_t = \kappa \Delta u, \quad u(x, y, 0) = 1$$

with NEUMANN boundary conditions

$$\nabla_{\mathbf{n}} u(1, y, t) = \nabla_{\mathbf{n}} u(-1, y, t) = \nabla_{\mathbf{n}} u(x, 1, t) = 0$$

and a NEWTON-type BC on the south side,

$$\nabla_{\mathbf{n}} u(x, -1, t) + \gamma u(x, -1, t) = 0,$$

a series solution can be obtained [4] by means of eigenvalues and eigenfunctions of the negative Laplacian:

$$u(x, y, t) = \sum_{j=1}^{\infty} \frac{2}{\xi_j} \frac{\sin \xi_j}{1 + \frac{2\gamma}{4\gamma^2 + \xi_j^2}} \cos \frac{\xi_j(y-1)}{2} e^{-\frac{\kappa \xi_j^2}{4} t},$$

where (for $\gamma > 0$) the ξ_j are solutions to $\tan \xi = 2\gamma/\xi$. This rapidly converging series ($t > 0$) is well suited for a comparison with simulation solutions on an $n \times n$ -grid

$$\left\{ \left(-1 + \frac{2(i-1)}{n-1}, -1 + \frac{2(j-1)}{n-1} \right); i, j = 1, \dots, n \right\}.$$

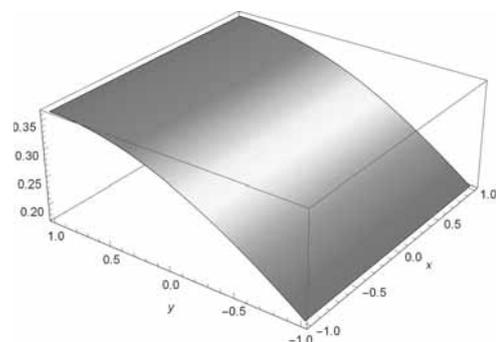


Figure 1: Analytical solution on Ω ($t = 4, \gamma = \kappa = 1$).



```
n=101; nx=ny=n; dx=2/(nx-1); dy=2/(ny-1); dx2=dx^2; dy2=dy^2; tend=5;
Z=Array[z,{nx,ny}]; X=Array[zreal,{nx,ny}]; Y=Array[zimag,{nx,ny}];
For[i=1,i<=nx,i++,For[j=1,j<=ny,j++,Z[[i,j]]=-1+(i-1)dx+I(-1+(j-1)dy)];];
func[t_][i_,j_]:=u[i,j][t]; U=Array[func[t],{nx,ny}]; X=Re[Z]; Y=Im[Z];
u[1,j][t_]:=u[2,j][t]; u[nx,j][t_]:=u[nx-1,j][t];
u[i,1][t_]:=u[i,2][t]/(1+dy); u[i,ny][t_]:=u[i,ny-1][t];
uxx[i_,j][t_]:= (u[i+1,j][t]-2u[i,j][t]+u[i-1,j][t])/dx2;
uyy[i_,j][t_]:= (u[i,j+1][t]-2u[i,j][t]+u[i,j-1][t])/dy2;
eqn=Table[u[i,j]'[t]==uxx[i,j][t]+uyy[i,j][t],{i,2,nx-1},{j,2,ny-1}];
ic=Table[u[i,j][0]==1,{i,2,nx-1},{j,2,ny-1}]; eqns=Flatten[Join[eqn,ic]];
vars=Flatten[Table[u[i,j][t],{i,2,(nx-1)},{j,2,(ny-1)}]];
result=NDSolve[eqns,vars,{t,0,tend},InterpolationPrecision->20,
Compiled->True,Method->{"EquationSimplification"->"Residual"}];
U=U/.result; lsg=Flatten[Table[{X[[i,j]],Y[[i,j]]},
SetAccuracy[U[[1,i,j]]/. t->4,50]},{i,1,n},{j,1,n}],1];
```

2 Simulation Solution on Square

On a computational grid like above, functions $u_{ij}(t)$ are now taking over and are being considered in each grid point (x_i, y_j) . Replacing the Laplacian and the boundary conditions with their discrete approximations yields a consistent system of ODEs to the original problem. The boxed Mathematica code demonstrates a crude but fully functional implementation that we shall call *FDI*. Note that two-point forward/backward boundary approximations and a five-star for the Laplacian in the interior points only produce a first-order convergent algorithm.

Finite difference formulae. Let us therefore first look out for higher order finite-difference approximations to the spacial derivatives involved. SCHIESSER[6] provides some of them and according to FORNBERG[1], his method is smartly being generalized to our two-dimensional needs: Suppose we are interested in finding the approximation

$$\frac{\partial^{\mu+v} u(x,y)}{\partial^\mu x \partial^\nu y} \approx \sum_{m=-m_0}^{m_1-m_0} \sum_{n=-n_0}^{n_1-n_0} a_{mn} u(x+m\Delta_x, y+n\Delta_y)$$

for a (potentially mixed) partial derivative of orders μ and ν in x - and y -direction. For generally given grid-

points let now Mathematica develop

$$\frac{\xi^{m_0} \eta^{n_0} (\ln \xi)^\mu (\ln \eta)^\nu}{\Delta_x^\mu \Delta_y^\nu} \approx \sum_{m=0}^{m_1} \sum_{n=0}^{n_1} a_{m-m_0, n-n_0} \xi^m \eta^n$$

in the Taylor series at $(\xi = 1, \eta = 1)$ to find the weights a_{mn} for the approximation in a specific gridpoint. Equidistance between the gridpoints is supposed but with m_0, m_1, n_0, n_1 we are able to claim how many neighbour points to the left/right and above/below should be incorporated which is necessary to specifically handle boundary or boundary-near points, depending on the desired order of convergence (expand to the proper Taylor polynomial). That is, m_0 is the count of intervals between the utmost left point and the point of interest whereas m_1 counts the intervals between the utmost points and similar for n_0, n_1 in y -direction.

For example, if we are interested in a fourth-order approximation of the x - derivative in a gridpoint lying on the left side of our unit square, we set $\mu = 1, \nu = 0, m_1 = 4$ and $m_0 = n_0 = n_1 = 0$ to find

$$\frac{\ln \xi}{\Delta_x} \approx \frac{1}{12\Delta_x} (-25 + 48\xi - 36\xi^2 + 16\xi^3 - 3\xi^4)$$

which means that at the left boundary we can change the first-order approximations in our code to

$$u' \approx (-3u_5 + 16u_4 - 36u_3 + 48u_2 - 25u_1)/12\Delta_x.$$

The other boundary sides are treated very similar and with $u' = 0$ or $u' = \gamma u_1$ (take in the latter case care of the derivative orientation) we solve this for u_1 to obtain a fourth-order accurate boundary approximation. It is clear that to obtain a global fourth order convergent algorithm, interior points for the Laplacian have to be treated in an appropriate way. Taking this into consideration, for $i = 3, \dots, n-2$ we make use of

$$u_{xx}(x_i, y_j, t) \approx \frac{-u_{i+2} + 16u_{i+1} - 30u_i + 16u_{i-1} - u_{i-2}}{12\Delta^2},$$

whereas in case of $i = 2$ (cases $i = n-1$ and u_{yy} analog)

$$u_{xx}(x_i, y_j, t) \approx \frac{u_6 - 6u_5 + 14u_4 - 4u_3 - 15u_2 + 10u_1}{12\Delta^2}$$

can be deployed. Replace the blue/bf highlighted code and you are ready to run a higher order code, *FD2*.

Results. With the presented series solution, we are able to validate the implementation of our CTDS-Code and also compare it against the results obtained by MATLAB's PDE-Toolbox. Let $\hat{u}(x, y, t)$ be the series solution on \mathfrak{Q} , then by considering the relative error (cf. Figure 2, Figure 3) in a grid point (x_i, y_j) ,

$$e_{i,j}(t) := \frac{\hat{u}(x_i, y_j, t) - u(x_i, y_j, t)}{\hat{u}(x_i, y_j, t)},$$

one has a tool for such a comparison at hand. Another possible way for checking the results is regarding the absolute errors $E_{i,j}(t) := |u(x_i, y_j, t) - \hat{u}(x_i, y_j, t)|$, see Figure 4, or their arithmetic average,

$$E(t) := \frac{1}{n^2} \sum_{i,j} E_{i,j}(t).$$

Using different sized grid spacings on \mathfrak{Q} (e.g. 51×51 , 101×101 and 201×201 is a good idea) by halving Δ also gives a clue about orders of convergence. $\mathcal{O}(\Delta)$ and $\mathcal{O}(\Delta^4)$ are as expected for *FD1* and *FD2* while *PdeTbx* shows up quadratically, cf. Table 1.

As a conclusion, major benefits in applying CTDS-methods are that orders of convergence can easily be adjusted and higher order difference formulae show up with formidable error behaviour even on coarse grids which helps to save computational time. Furthermore, replacing spatial derivatives by discrete differences gives rise to treatment of more general classes of PDEs while *PdeTbx* has its limitations in this regard.

A severe limitation is that our method is bound to orthogonal grids (and working best on equidistant grid sizes). We focus on that by regarding conformal maps.

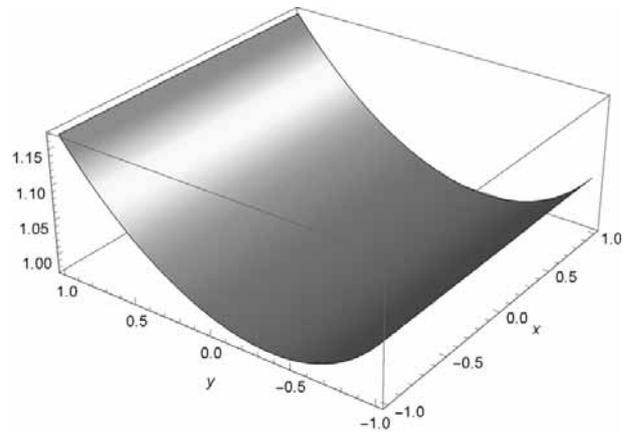


Figure 2: Rel. errors $e_{i,j}(4)$ on \mathfrak{Q} for $n = 101$, method *FD1*.

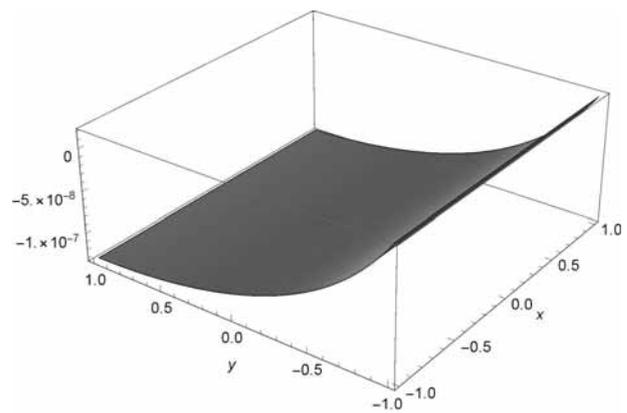
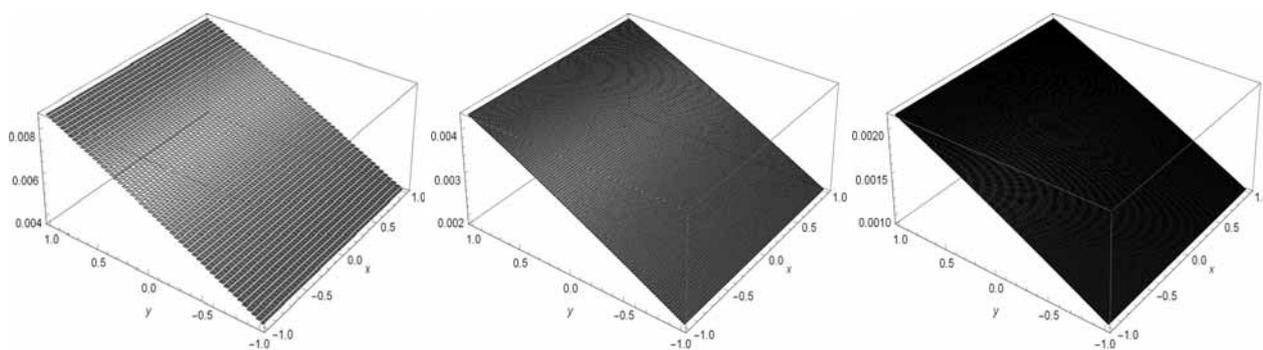


Figure 3: Rel. errors $e_{i,j}(4)$ on \mathfrak{Q} for $n = 101$, method *FD2*.

Method	Points	Avg. Abs. Error	Calc. Order
<i>FD1</i>	2.601	0.00627	—
<i>FD1</i>	10.201	0.00312	2.00717
<i>FD1</i>	40.401	0.00156	2.00360
<i>FD2</i>	2.601	$4.67382 \cdot 10^{-9}$	—
<i>FD2</i>	10.201	$2.99177 \cdot 10^{-10}$	15.62228
<i>FD2</i>	40.401	$1.82074 \cdot 10^{-11}$	16.43158
<i>PdeTbx</i>	2.577	$3.47640 \cdot 10^{-6}$	—
<i>PdeTbx</i>	10.145	$8.64349 \cdot 10^{-7}$	4.02199
<i>PdeTbx</i>	40.257	$2.15899 \cdot 10^{-7}$	4.00349

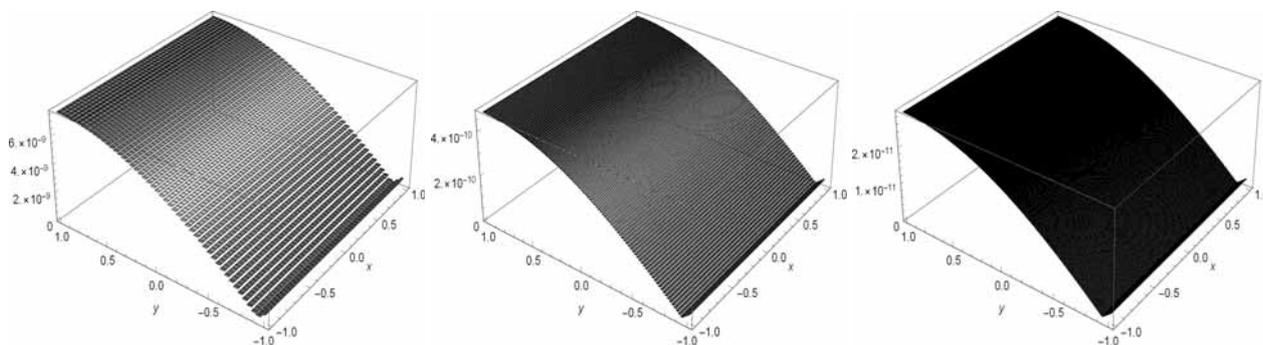
Table 1: Comparing mean arithmetic absolute errors, $t = 4$.



FD1 2.601 Pt

FD1 10.201 Pt

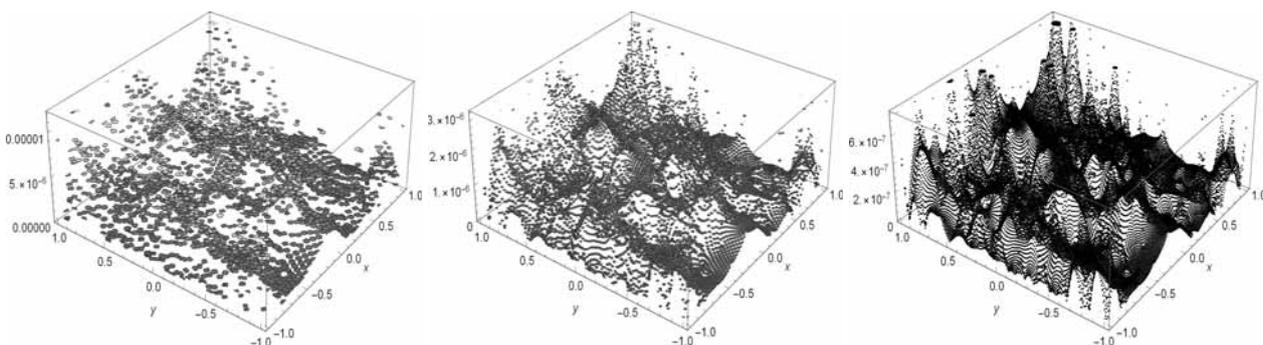
FD1 40.401 Pt



FD2 2.601 Pt

FD2 10.201 Pt

FD2 40.401 Pt



PdeTbx 2.577 Pt

PdeTbx 10.145 Pt

PdeTbx 40.257 Pt

Figure 4: Absolute errors $E_{i,j}(4)$ for different grid sizes on Ω .

3 Conformal Transformations

Conformal maps preserve angles between intersecting grid lines. Moreover, using cartesian co-ordinates on \mathfrak{Q} , local base vectors in a specific grid point are rotated but stretched by the same factor by such a transformation, $f(z)$. To construct the map itself, we argue by means of a *point transformation* of the complex plane while mapping the square to a given domain (demonstrated with the unit disk, \mathfrak{E}) will later on be interpreted as a *transformation of co-ordinates* to simulate a PDE on this domain. Under this point of view the given domain is being parameterized by the unit square.

Regarding our heat equation, the Laplacian is not invariant but transforms in a very convenient way, merely a (from point to point differing) factor $1/|f'(z)|^2$ appears to correct the impact of the new curvilinear but orthogonal co-ordinates. Keeping this in mind, we compute the map from \mathfrak{Q} to \mathfrak{E} , save its derivative values in each gridpoint and use them in a simulation study on \mathfrak{E} .

Back in about 1869 H.A. SCHWARZ and E.B. CHRISTOFFEL independently found that the upper complex plane $\Im(w) > 0$ (or \mathfrak{E} respectively, just consider a Moebius transformation) can be conformally mapped to a polygonal arc with vertices z_k by elliptic integrals [2]

$$z = f(w) = C + D \int_0^w \prod_{k=1}^n (\zeta - w_k)^{-\beta_k} d\zeta.$$

The constants C and D act as translation and rotation/stretching respectively, to map \mathfrak{E} to \mathfrak{Q} we can fix them immediately demanding $f(0) = 0$ and $f(1) = 1$. With $\pi\beta_k$ denoting the outer tangent turning angle in vertex z_k , we determine $\beta_k = 1/2$ for all four square vertices and the vertex preimages w_k can in this case be prescribed by $w_{k+1} = \exp[i(\frac{\pi}{4} + \frac{k\pi}{2})]$, $k = 0, \dots, 3$. Note that for more general polygonal arcs finding the preimages can be a tough task [7]. Putting all together,

$$z = f(w) = -\frac{2\sqrt[4]{-1}}{\mathfrak{E}_{k;\frac{1}{2}}} \mathfrak{E}_f \left[i \operatorname{arsinh} \left(\sqrt[4]{-1} w \right); -1 \right]$$

maps the disk to our unit square where elliptic functions help us to express the map in closed form. As f is bi-holomorphic and we are interested in the inverse transformation from square to disk, Mathematica handles this by the definition $-(-1)^{1/4} \operatorname{JacobiSN}[z * \operatorname{EllipticF}[I * \operatorname{ArcSinh}[(-1)^{1/4}], -1], -1]$. Obviously not only the conformal map but also its derivatives can be evaluated to any desired precision.

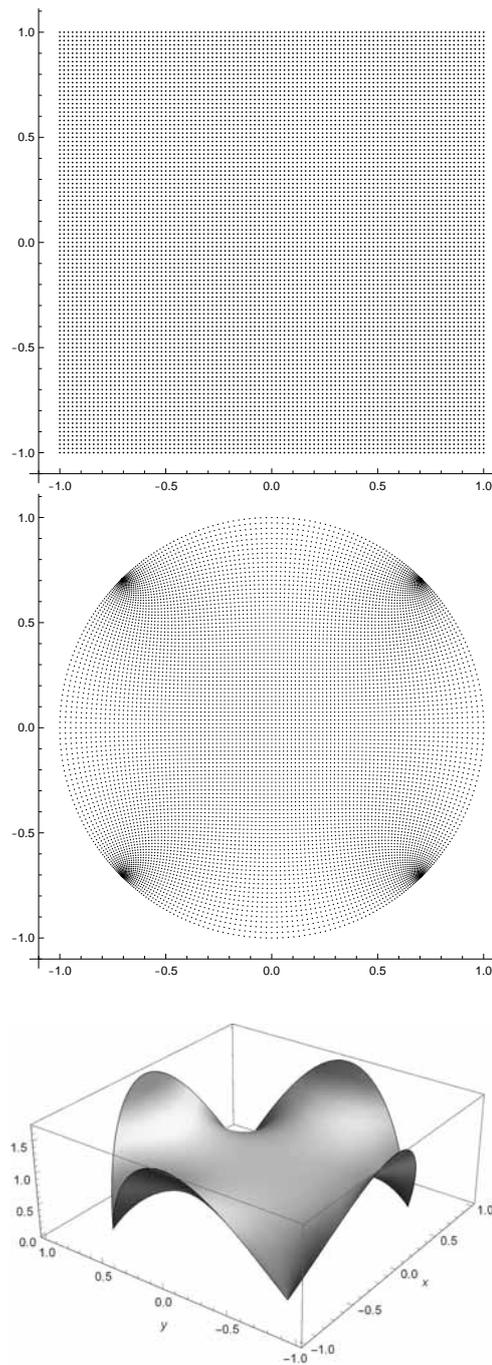


Figure 5: Computational grid ($n = 101$) before (t) and after (c) transformation with $|f'(x,y)|^2$ shown at the bottom. The conformal map from square to disk can also be interpreted as an introduction of curvilinear co-ordinates on \mathfrak{E} which changes the fundamental tensor from $\delta_{ij} \rightarrow |f'(z)|^2 \delta_{ij}$.

4 Series Solution on Unit Disk

Let us now consider a heat conduction problem on \mathfrak{E} :

$$u_t = \kappa \Delta u, \quad u(\xi, \eta, 0) = 1 - (\xi^2 + \eta^2)$$

with DIRICHLET boundary condition

$$u(\xi, \eta, t) = 0, \quad |\zeta| = 1 \quad \forall \zeta = \xi + i\eta.$$

Similar to the problem on \mathfrak{Q} a (symmetric) solution on \mathfrak{E} can be derived by means of BESSEL functions [3]

$$u(r, t) = 4 \sum_{k=1}^{\infty} \frac{e^{-\kappa t \lambda_{0,k}^2} \mathfrak{J}_0(r \lambda_{0,k}) \mathfrak{J}_2(\lambda_{0,k})}{\lambda_{0,k}^2 (\mathfrak{J}_0(\lambda_{0,k})^2 + \mathfrak{J}_1(\lambda_{0,k})^2)}$$

with zeroes $\lambda_{0,k}$ of $\mathfrak{J}_0(x)$ and (for $t > 0$) staggering convergence behavior, thus well suited for comparison with simulation results (Figure 6). If you do not like to calculate by hand, in case of our specific problem you can even let Mathematica do the job¹ for you.

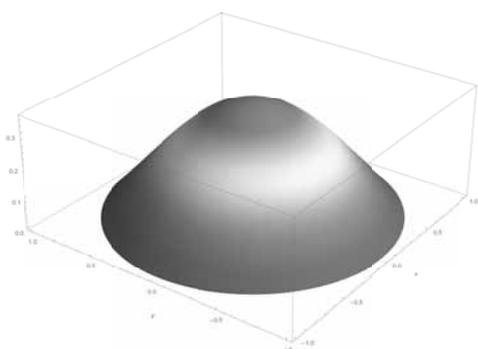


Figure 6: Analytical solution on \mathfrak{E} ($t = 2, \kappa = 1/10$).

5 Simulation Solution on Disk

To get the simulation of the disk problem ready to run, one merely has to numerically calculate the first derivatives $f'(z) = \xi_x + i\eta_x$ of the conformal transformation on any considered grid to the desired precision, save them as a list on the grid and merely replace the Lapla-

¹ eqn = r D[u[r,t],t,1] == κ D[rD[u[r,t],r],r];
 bc = u[1,t] == 0; ic = u[r,0] == 1-r^2;
 dsol=DSolve[{eqn,bc,ic},u[r,t],{r,t}]

cian on the righthandside of the equation:

$$\Delta u \rightarrow \frac{\Delta u}{\xi_x^2 + \eta_x^2}.$$

Obviously our boxed Mathematica code is very easy to adapt, we just have to modify the *bf/italic* expression. Note that the singularities arising in the vertex points caused by the map can be neglected since values at these points are not being used in the simulation run and must not be provided. This argument is given with respect to transformed Newton boundary conditions.

As for the pre-given Dirichlet boundary conditions, it turns out that convergence order of *FD1* increases to be quadratic, by replacing the first section of the blue-colored code dealing with the boundary conditions by assignment of fixed zero values our simulation code is ready to run and comparable to the series solution.

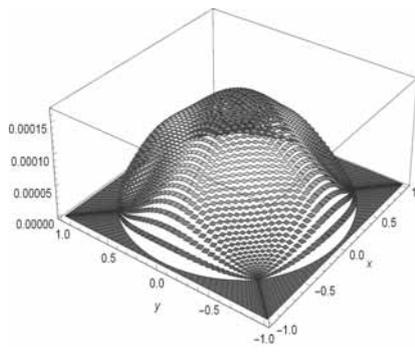
For a valid comparison with *PdeTbx* results, it is now necessary to focus on grid sizes with a similar amount of grid points and for that reason the grids $46 \times 46, 91 \times 91$ and 181×181 seem to be appropriate.

Table 2 shows that even algorithm *FD1* performs slightly better than *PdeTbx* with respect to averaged absolute errors. As for *FD2*, note that forth order convergence is not achieved when switching from $46 \times 46 \rightarrow 91 \times 91$ grid but only shows up with a factor 5.7 instead of about 16 - we suspect this is essentially caused by the fact that in this case of an even number of gridpoints the origin point is not being used in the calculations. Overall absolute errors on disk are shown in Figure 7.

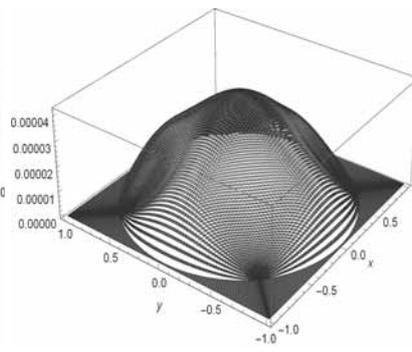
For our specific heat conduction problem on unit disk it enlightens that application of conformal transformation had no impact of error or convergence behavior (numerical values calculated up to 50 digits precision).

Method	Points	Avg. Abs. Error	Calc. Order
<i>FD1</i>	2.116	$6.27281 \cdot 10^{-5}$	—
<i>FD1</i>	8.281	$1.61182 \cdot 10^{-5}$	3.89176
<i>FD1</i>	32.761	$4.08124 \cdot 10^{-6}$	3.94933
<i>FD2</i>	2.116	$2.58897 \cdot 10^{-7}$	—
<i>FD2</i>	8.281	$4.55414 \cdot 10^{-8}$	5.68487
<i>FD2</i>	32.761	$2.83388 \cdot 10^{-9}$	16.07035
<i>PdeTbx</i>	2.129	$1.48329 \cdot 10^{-4}$	—
<i>PdeTbx</i>	8.385	$3.76648 \cdot 10^{-5}$	3.93812
<i>PdeTbx</i>	33.281	$9.48946 \cdot 10^{-6}$	3.96912

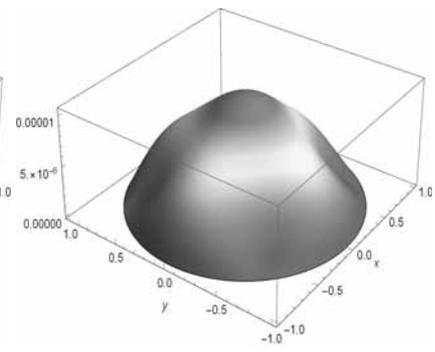
Table 2: Error comparison on disk. Quadratic order for *FD1*, non-optimal starting grid for *FD2*.



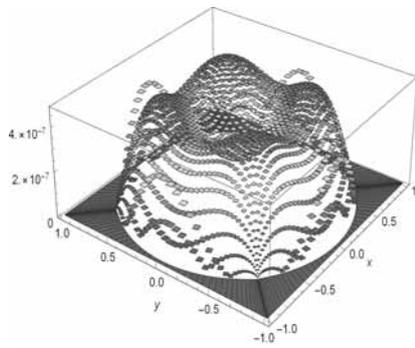
FD1 2.116 Pt



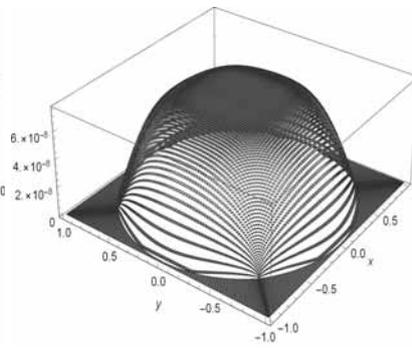
FD1 8.281 Pt



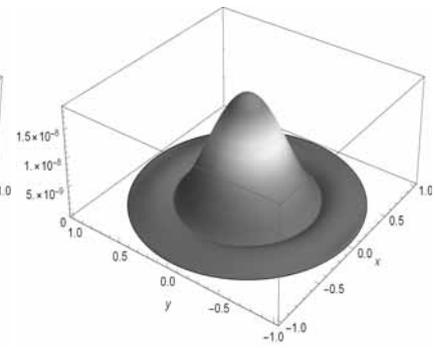
FD1 32.761 Pt



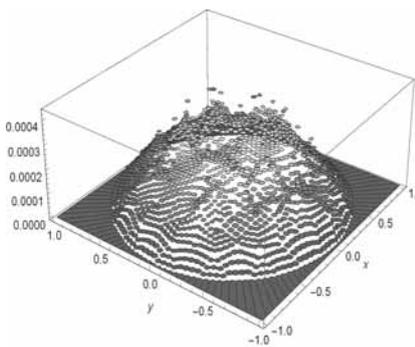
FD2 2.116 Pt



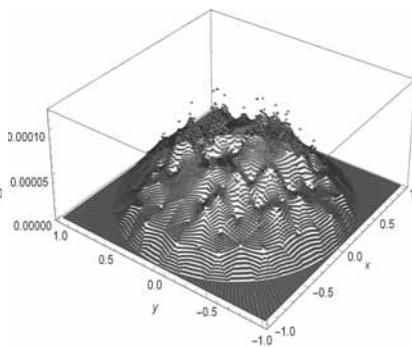
FD2 8.281 Pt



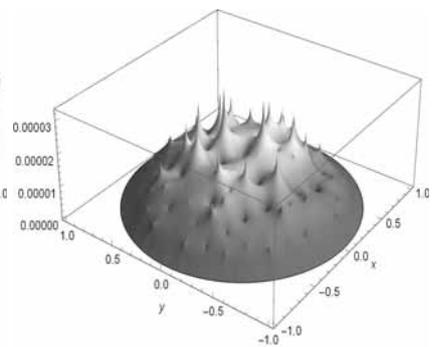
FD2 32.761 Pt



PdeTbx 2.129 Pt



PdeTbx 8.385 Pt



PdeTbx 33.281 Pt

Figure 7: Absolute errors $E_{i,j}(2)$ for different grid sizes on \mathbb{E} .

Summary

The results of this paper show that combining the method of lines with conformal mappings is principally apt for numerical and/or symbolical treatment of PDEs. Moreover, the analytic solutions to problems on unit square and unit disk presented in this article are convenient for checking your own simulation code.

The developed simulation code in Mathematica can be regarded as a starting point from which one is able to march out and adapt it to his own needs: In keeping things modular, a diversity of higher-order derivative approximations can externally be implemented and loaded according to the desired error requirements (see the blue/bf coded section in the box).

On the other hand, a very broad range of PDE-classes are accessible by this approach (for time-dependent problems the bf/italic coded section is to be modified). While even tensor-valued state variables can be considered, the presented method is definitely restricted to two-dimensional geometries. This is caused by the fact that conformal transformations arise in complex analysis mapping $C \rightarrow C$.

So if we are able to construct a highly accurate conformal map from square to a more general domain [5] (aside the unit disk but potentially for convenience with the unit disk interconnected) and same holds for the numerical values of its derivatives, we would be in a good position to treat PDEs on such domains as well. This will be shown in a subsequent contribution.

In addition, another level of abstraction will be introduced by regarding physical laws in invariant PDE-formulation. The domains under consideration will then appear as two-dimensional (flat) Riemannian manifolds where the calculated derivatives of the conformal map are hence to be interpreted as metric quantities correcting the effect of using curvilinear co-ordinates: Fundamental tensor and Christoffel symbols will be used to realize the PDE in this 'conformal co-ordinates'.

In the presented method the conformal map played a double game: Grasping it as a point map we constructed the transformation from square to disk. On the other hand, to get a glimpse at the metric conditions, the map acted as transformation of co-ordinates supporting simulation - the disk has been parameterized by the square.

References

- [1] Fornberg B. *Calculation of Weights in Finite Difference Formulas*. SIAM Rev., Vol. 40, No. 3, pp. 685-691, 1998, ISSN (print): 0036-1445.
- [2] Hassenpflug WC. *Elliptic Integrals and the Schwarz-Christoffel Transformation*. Computers & Mathematics with Applications, Vol. 33. 15-114. 10.1016/S0898-1221(97)00091-6.
- [3] Herod JV. *Asynchronous Studies in Undergraduate Partial Differential Equations. Lecture 33: The Heat Equation on a Disk*. <https://people.math.gatech.edu/~herod/conted/M33.pdf>
- [4] Holzinger M. *Konforme Abbildungen zur Simulation von Modellen mit verteilten Parametern*. Dissertation/PhD (in German), Wien, 2020. <http://katalog.ub.tuwien.ac.at/AC15652638>
- [5] Kythe PK. *Computational Conformal Mapping*. Birkhäuser, Boston 1998, ISBN 0-8176-3996-9.
- [6] Schiesser WE. *The Numerical Method Of Lines. Integration of PDEs*. Academic Press Inc., San Diego, 1991, ISBN 0-12-624130-9.
- [7] Trefethen LN, Driscoll TA. *Schwarz-Christoffel mapping in the Computer Era*. Proceedings International Congress of Mathematicians, Vol. III (Berlin, 1998), vol. 1998, pp. 533-542, ISSN 1431-0643.

Timed Discrete-Event Simulation of Aviation Scenarios

Hai Nguyen Van^{*}, Frédéric Boulanger^{**}, Burkhart Wolff^{**}

Université Paris-Saclay, CNRS, LRI, 91405 Orsay, France; ^{*}ORCID 0000-0002-0585-1651; ^{**}firstname.lastname@lri.fr

SNE 30(2), 2020, 51-60, DOI: 10.11128/sne.30.tn.10512
Received: February 2, 2020; Revised April 25, 2020;
Accepted: May 2, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Ensuring systems behave as they are expected is unavoidable in the context of critical environments. In the aviation industry, certification standards provide rules and protocols to ensure correct maneuvers with respect to logical or timed events. These are targeted to computer-intensive systems as well as to human flight crews. In this setting, we are interested in the modeling and simulation of event-driven and time-driven behaviors at a high level. This study focuses on the TESL language [1] that provides a logical framework for timed behaviors with monitoring and testing features. In particular, we model various aviation scenarios and focus our study on fault monitoring.

Introduction

In past years, an increase in modeling and simulation in industry has emerged to assist engineers and designers of various process levels. In a broader way, this has been ensured by the emergence of Model-Based Design that allows the differentiation of stages and components composing large systems. These large systems consist of modeled components of various nature and form a multi-paradigm environment where each part is modeled with its own semantics of execution: this is commonly called *heterogeneous modeling* [2]. For instance in control systems, mode switches can be modeled with finite-state machines, and sensor data processors with dataflow models. Recent advances have proved that these submodels could be unified to form a *supermodel*. Figure 1 highlights this idea where each submodel is described by a different paradigm, then they are coordinated as a supermodel.

Complementary to modeling, the increased demand of automatic validation relates to the critically-large

models where sole human analysis no longer suffices. The addition of mathematics and logics to the understanding of modeling and computing problems at a larger scale is named *formal methods*. The problem is two-fold. On one side, heterogeneous modeling raises the question of the adaptability of paradigms. Indeed, each modeling paradigm comes with a specific model of computation detailing a precise semantics of execution for each submodel. On the other side, validation demands a unified environment for safety property verification or test generation for these various paradigms. In the last decades, several multi-paradigm frameworks have appeared and attempt to address these issues, *e.g.*, Ptolemy II [3, 4], ModHel'X [5], BCool [6].

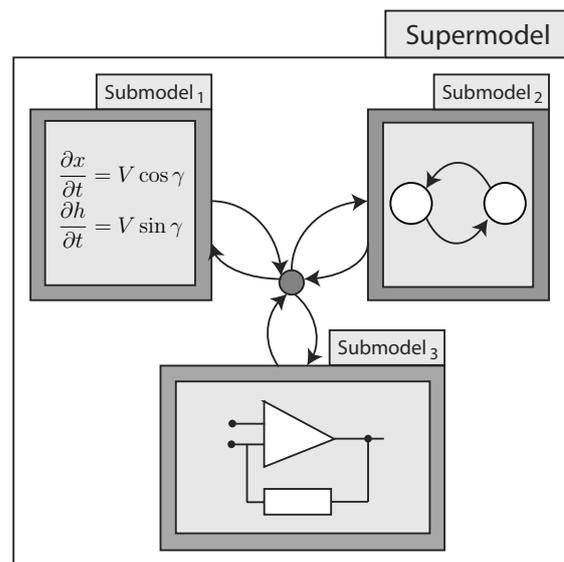


Figure 1: A supermodel for heterogeneous submodels

Our study focuses on TESL [1] which has been introduced as the inner language of the ModHel'X framework. Similarly to intermediate programming languages in compiler theory, models in ModHel'X are coordinated using TESL. Indeed, it is a specification language that describes discrete-events with time anno-

tations (*tags*). This flavor of chronometric time is necessary to compose modeled systems where events are described with respect to chronometric time durations, instead of logical time as would be found in temporal logics [7].

In this paper, we highlight the application of this language to aeronautical systems. We are interested in modeling standard scenarios used by common aircrafts, and validated by airworthiness authorities (ICAO Annex 8 [8], EASA AIR OPS [9]). Our goal is to exhibit a unified modeling framework suited for validation and verification by means of:

- multi-level model specification;
- generation of execution traces for simulation purposes;
- real-time testing and system monitoring.

In the heterogeneous context, our framework allows to abstract from programming details in order to reason on high-level behaviors. We believe this is particularly useful for aviation-related systems. Indeed, software in current airborne systems are usually certified for the highest Design Assurance Level as defined by RTCA/DO-178C [10, 11]. This certification is known to provide guidance for software life-cycle processes, and emphasizes on verifying the relation between high-level requirements and low-level implementation. The testing topic of this certification is especially focused on requirements. Our work precisely targets this problem by providing a specification-based testing/monitoring framework. For these reasons, we believe our study addresses several current Flight Control Systems (FCS):

Multi-pilot aircrafts. The Airbus A320 aircraft family is a well-known system made of fault-tolerant components based on redundancy and dissimilarity. Traverse [12, 13] reported that the aircraft primary control surface computers were designed by different design analysts on independent architectures. This case of heterogeneity clearly exhibits how a multi-paradigm environment needs to be unified in order to be validated at a higher-level.

Unmanned aircrafts. Furthermore, recent advances in unmanned aircrafts [14, 15] make the topic of verification and validation even more crucial due to the need of safe, reliable and fully automated software-intensive systems [16, 17] where, accordingly, design and test procedures tend to be fully automated.

1 The TESL Language

The core language of our study is the TESL language. It is inspired from the CCSL language [18, 19] and the Tagged Signal Model [20]. It lies at the heart of the ModHel'X modeling and simulation framework and serves as an intermediate representation for simulation solvers. In our setting, events are described and specified by *clocks*. A clock that *ticks* means that the associated event is occurring. These entities are ruled by three kinds of modality:

- *Event-driven implications.* An occurring event can trigger another one: “If clock K_1 ticks, then clock K_2 will tick under conditions”.
- *Time-driven implications.* An occurring event triggers another one after a chronometric time delay measured on the time scale of a specific clock. Remark that this delay is a duration expressed as a difference between two tags, and not as a number of ticks.
- *Tag relations.* By default, clocks live in independent *time islands*. The purpose of tag relations is to link these different time scales, *e.g.*, time expressed in seconds and minutes admits an arithmetic relation stating that time flows 60 times as fast in seconds than in minutes. This does not mean that the seconds clock ticks 60 times more than the minutes clock, but simply that their tag annotation satisfies this arithmetic relation.

To provide a glimpse of the core features of TESL, we present a brief grammar of the language:

- *tag relation* $[K_1, K_2] \in R$
The time frames of clocks K_1 and K_2 are related by the arithmetic relation R .
- *K_{evt} sporadic τ on K_{meas}*
Some event will occur on clock K_{evt} at timestamp τ measured on clock K_{meas} .
- *K_{master} implies K_{slave}*
At every instant, if K_{master} ticks, then K_{slave} instantaneously ticks.
- *K_{master} sustained from K_{begin} to K_{end} implies K_{slave}*
In the interval between a tick on clock K_{begin} and a tick on clock K_{end} , K_{master} implies K_{slave} as previously (scoped implication).

- K_{master} time delayed by τ on K_{meas} implies K_{slave}
Whenever the master clock K_{master} ticks, the time tag on the measuring clock K_{meas} is measured and delayed by duration τ to yield the date of a future instant at which clock K_{slave} will tick.
- K_1 strictly precedes K_2
Any event occurring on K_2 is preceded in the strict past by a distinct event occurring on K_1 .

2 The Takeoff Scenario



Figure 2: Airspeed indicator and altimeter (courtesy of Laminar Research)

To illustrate how time scales can be constructed in TESL, we are interested in modeling the takeoff procedure of a small single engine aircraft with basic parameters: time, airspeed and altitude. We assume usual atmospheric conditions and specifically chose to model performance parameters extracted from the Cessna 172 aircraft [21, 22]. The rotation speed V_R specifies when the pilot should move the pitch control backwards to generate lift.

Speed	Description
$V_R = 55 \text{ kt}$	Rotation speed

Table 1: Extract of V-speeds of the Cessna 172

In the next subsections, we first introduce an execution trace to provide intuitions for our case study. Then, we will exhibit the TESL specification for this scenario.

2.1 Clocks and execution traces

To define the basic quantities and events in which we are interested, we define *clocks* that describe the timeline of events. These are embedded with time tag an-

notations that rule how quantities and units are related. Each clock denotes a quantity with a specific unit:

```
// Declaring quantities and units
rational-clock time-S           // in [s]
rational-clock time-MIN        // in [min]
rational-clock speed-MPS       // in [m.s-1]
rational-clock speed-KT        // in [kt]
rational-clock altitude-FT     // in [ft]
```

Moreover, the specification is augmented with three clocks for the events of our interest:

- VR-reach: speed reaches V_R ;
- liftoff: the aircraft is airborne;
- flaps-retract: flaps are retracted.

The TESL language is a specification language that allows to describe traces. Figure 3 depicts a minimal execution trace with three instants. At the first instant, time is just 0 s. Then at the second instant, speed has reached $V_R = 55 \text{ kt}$ at 12.2 s. Hence, clock VR-reach is triggered, and so is liftoff consequently: they are said to be *ticking synchronously*. Then at the third instant, flaps-retract ticks at the altitude of 400 ft at 27.2 s.

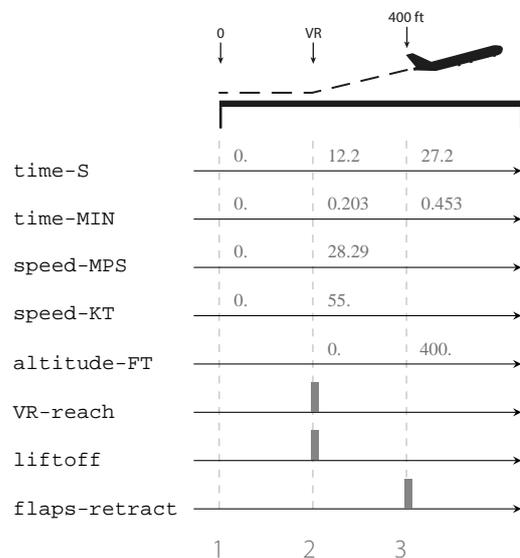


Figure 3: Execution trace when performing takeoff

Remark. The above trace depicts a minimal run. It illustrates an observation. Other instants may exist in between those mentioned, where none of our events occur. These instants are simply not “observed”.

2.2 Causality and Timestamp Relations

Units In the next paragraphs, we use TESL to describe the above potential behavior.

```
// Unit conversion between [s] and [min]
tag relation time-S = 60.0 * time-MIN
// ... and between [m.s-1] and [kt]
tag relation speed-KT = <sup>3600</sup>/1852 * speed-MPS
```

Clocks `time-S` and `time-MIN` respectively express time given in seconds and minutes. Clocks `speed-MPS` and `speed-KT` respectively denote speeds in meters per second and knots (as given by the airspeed indicator). We use tag relations to describe unit conversions between such quantities. In particular, tags on clocks `time-S` and `time-MIN` shall satisfy the arithmetic relation that one minute is equivalent to sixty seconds. Likewise, one knot is equivalent to $\frac{3600}{1852} \text{ m} \cdot \text{sec}^{-1}$. In our context, the notion of time lies under tag annotations. Compared to temporal logics where time is purely logical, we precisely capture chronometric durations.

Acceleration and liftoff Tag relations also allow to describe how quantities are related and can define the acceleration profile of the modeled aircraft. Here we consider the uniform acceleration of a light aircraft gaining speed at $4.5 \text{ kt} \cdot \text{sec}^{-1}$:

```
tag relation speed-KT = 4.5 * time-S
```

Remark. The permissive nature of the language also allows to leave this unspecified. We could have also required to design a general-purpose specification independently from the physical profile.

Whenever speed reaches V_R at 55 kt, the event `VR-reach` is triggered, and instantaneously triggers `liftoff` indicating that the aircraft is airborne.

```
VR-reach sporadic 55.0 on speed-KT
VR-reach implies liftoff
```

Flaps retraction To quickly reach the desired altitude, the pilot controls the pitch (longitudinal axis) to maintain the airspeed at fixed value V_y while climbing. This approximately corresponds to a vertical speed of $1200 \text{ ft} \cdot \text{min}^{-1}$. The relation is written:

```
tag relation altitude-FT =
    1200.0 * time-MIN + -244
```

Finally, the aircraft reaches the altitude for flaps retraction at 400 ft.

```
liftoff time delayed by 400. on altitude-FT
implies flaps-retract
```

3 The Autobrake System

In this section, we explore more complex specifications by employing a mix of event and time-triggered events, and the usage of sequential and asynchronous operators. We are interested in modeling the takeoff procedure of a transport-category aircraft with takeoff rejection components such as the Airbus A320 [24, 25]. As said earlier, decision, rotation and lift off do not necessarily coincide (depending on the aircraft performance category). As a matter of fact, the manufacturer distinguishes and specifies the following speed thresholds as illustrated in Table 2.

Speed	Description
$V_1 = 118 \text{ kt}$	Decision speed
$V_R = 126 \text{ kt}$	Rotation speed

Table 2: Extract of V-speeds of the A320 under given physical assumptions [26]

As depicted in Figure 4, decision speed V_1 defines the speed limit at which the pilot in command is allowed to reject takeoff (RTO). Then V_R is the rotation speed as previously described, and lift off occurs 3 s after. Should the pilot decide to reject after V_1 , the aircraft would brake on a too short remaining runway, and hence overrun.

3.1 Acceleration and Speed Thresholds

We define speed thresholds as previously and also add a precedence constraint. This emphasizes on the fact that reaching V_R must have been preceded by reaching V_1 prior.

```
V1-reach strictly precedes VR-reach
V1-reach sporadic 118.0 on speed-KT
VR-reach sporadic 126.0 on speed-KT
```

Again, to define a specification independently from the aircraft performance, the precedence operator is

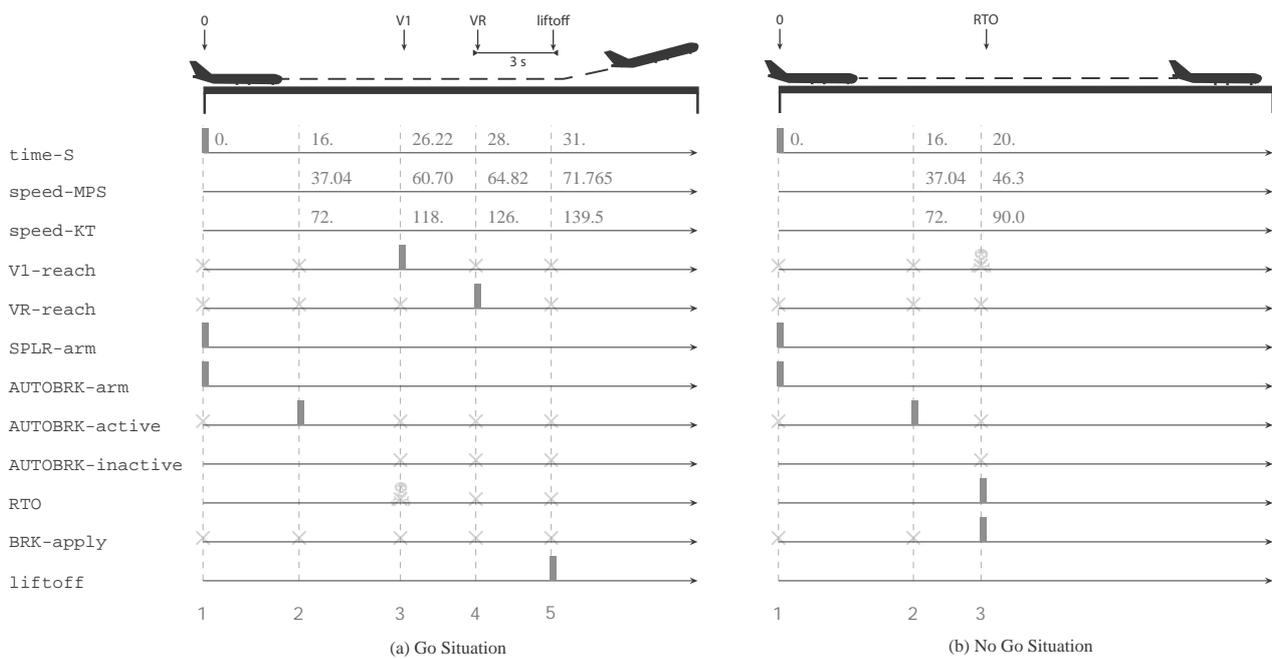


Figure 4: Execution trace when rejecting takeoff

enough and the two following lines could have been ignored. Finally, the aircraft is airborne 3 s after rotation speed has been reached.

```
VR-reach time delayed by 3. on time-S
                               implies liftoff
```

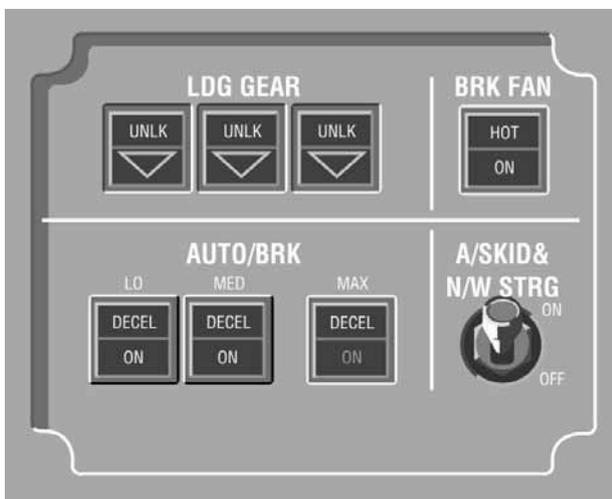


Figure 5: Autobrake command switch on panel board (extracted from [23])

3.2 Rejecting Takeoff with Autobrake

One of the aircraft braking systems is named *Autobrake* [24], and opposes to manual pedal braking. Its usage is preferable as:

- The number of brake pressure is minimized, reducing brake wear;
- A symmetrical brake pressure is applied ensuring an equal braking effect on gear wheels, especially on wet runways.

The system is activated whenever the following conditions are met:

1. Ground spoilers ARMED
2. Auto brake ARMED
3. Speed exceeds 72 kt
4. Accelerate-stop with THRUST on IDLE

These above conditions are written as:

```
SPLR-arm strictly precedes AUTOBRK-active
AUTOBRK-arm strictly precedes AUTOBRK-active
AUTOBRK-active sporadic 72.0 on speed-KT
```

Finally in the event of takeoff rejection (RTO), the system applies brakes if autobrakes were in an active state.

```
RTO sustained from AUTOBRK-active
to AUTOBRK-inactive implies BRK-apply
```

Event Death Killing a clock allows to prevent it from ticking, *i.e.* from having further event occurrences. This mechanism can be used to describe different forms of race conditions. Likewise, we can specify that reaching V_1 will prevent the event of rejecting takeoff. Conversely, applying brakes prevents from reaching V_1 .

```
V1-reach kills RTO
BRK-apply kills V1-reach
```

3.3 Simulation

Two situations that satisfy the above specification are depicted in Figure 4. The first execution trace in Figure 4a shows the Go situation where the pilot normally proceeds to aircraft takeoff. In the first instant, the pilot arms ground spoilers and autobrake. Consequently, autobrake is activated when speed exceeds 72 kt. There is no takeoff rejection (clock RTO), and the aircraft keeps on accelerating by reaching V_1 and V_R speeds until being airborne (clock `liftoff`) after a delay of 3s.

On the other side, a No Go situation is illustrated in Figure 4b where speed has exceeded 72 kt but takeoff rejection has been declared at 20s when the speed is 90kt, which immediately triggers brakes. This prevents from reaching V_1 and consequently V_R and aircraft liftoff.

4 Towards Hybrid Systems: Accelerate-Stop Distance

Our language also addresses the modeling of hybrid systems. This is exhibited by the ability to define differential equations in tag relations. In our case study, we can refine our specification to take into account differential quantities. In the case of a takeoff rejection, it

is necessary to ensure that the two-stage acceleration-deceleration ensures that the aircraft does not overrun, and remains within the limits of the runway. To provide a good abstraction level to the reader, we chose to deliberately simplify the definition of the Accelerate-Stop Distance and not take into account the recognition and decision time as specified by airworthiness authorities ([9], CAT.POL.A.205 Take-off).

Acceleration To compute these distances, we need to express instantaneous distance with respect to current time and speed. If we denote x as the distance of the running aircraft, provided time t and speed v , we have

$$dx = v \cdot dt$$

In our setting, this is straightforwardly expressed as

```
tag relation (d distance-M)
= speed-MPS * (d time-S)
```

where `distance-M` is the quantity denoting the run distance during acceleration phase.

Deceleration As done previously, we need to define how deceleration is expressed and accordingly the run distance. To proceed so, we will define new clocks for this deceleration stage: `speed-MPS-DECEL`, `speed-KT-DECEL` and `distance-M-DECEL`. Likewise, to keep our model simple enough but still relevant, we will assume a uniform deceleration of $-3\text{kt}\cdot\text{s}^{-1}$. In TESL, this would be expressible as a linear tag relation as previously, or equivalently with a differential equation between speed and time:

```
tag relation (d speed-KT-DECEL)
= -3.0 * (d time-S)
```

Finally, distance during deceleration relates to the following tag relation:

```
tag relation (d distance-M-DECEL)
= speed-MPS-DECEL * (d time-S)
```

To run our specification with a concrete example, let us assume that takeoff rejection has been declared at 20s. Figure 6 illustrates this process where the aircraft has approximately reached 520m at 20s. Finally, the aircraft reaches speed zero at 50s with a final run distance of approximately 1159m.

Remark. The precision of the differential calculus lies in the ODE solver in use. In this example, we

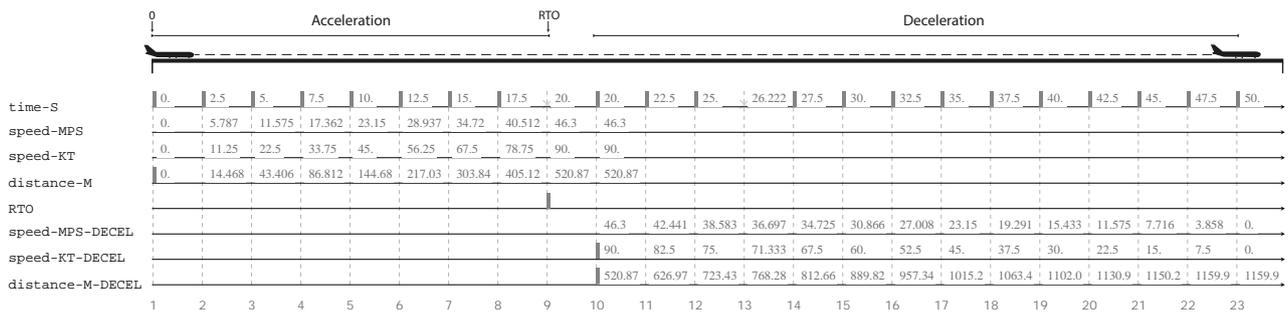


Figure 6: Execution trace when performing accelerate-stop

used the simple first-order forward Euler with an integration step of 2.5s. Our language is agnostic to this design choice and potentially admits any other relevant ODE solver. It aims at bridging gaps between different paradigms generally found for modeling complex systems.

5 Formal methods

5.1 Testing and Monitoring

In the previous sections, we have observed that the TESL language specifies execution traces. These can be constructively generated with a solver named Heron [27]. It is a multicore-aware solver made of approximately 4,000 lines of Standard ML [28] code compiled with MPL [29]/MLton [30]. It solves TESL specifications by exhaustively constructing execution traces as illustrated in the previous paragraphs. In particular, this allows to exhaustively test and monitor systems to the extent of the system observation interfaces. Our solver fetches observations provided by an external driver and filters out irrelevant execution trace branches. By keeping the only satisfying runs of the specification combined with those compliant with the observation of the system, the solver keeps exploring exhaustive possibilities. Whenever the solver faces an unwanted behavior, it will finally filter out all branches and remain in an inconsistent state, meaning that the system has produced a violating behavior.

For this purpose, it is possible to suggest a scenario to the solver and request a simulation trace (if it exists). From the specification of the Autobrake in Section 3 and its corresponding satisfying execution traces in Figure 4, we can specify an additional directive to the solver and request a run where the clock RTO is the *only*

one allowed to tick at instant 3. This is written

```
@scenario strict 3 RTO
```

The Heron solver will find no possibly satisfying run as it is not possible for RTO to tick alone without BRK-apply to tick as well. After successfully generating 2 instants, it will eventually fail and output the following:

```
##### Solve [1] #####
-> Consistent premodels: 1
-> Step solving time measured: 0.005 s
##### Solve [2] #####
-> Consistent premodels: 1
-> Step solving time measured: 0.016 s
##### Solve [3] #####
-> Consistent premodels: 0
-> Step solving time measured: 0.003 s
### ERROR: No further state found.
```

The solver, its source code and all mentioned examples in this paper are provided at github.com/heron-solver/heron.

5.2 Formalized Semantics

In an effort to fully validate our language and its logical foundations, a fragment of the TESL language, which consists of core formulae, has been proved to enjoy good and formal properties ensuring its well-foundedness. It has been formalized with two kinds of logical semantics providing an accurate meaning of language terms and how they are supposed to behave:

- a *denotational semantics* [31] mathematically describes the set of execution traces denoted by the language;
- an *operational semantics* [27] describes how the languages behaves/executes to generate traces.

These two semantics have been proved to be equivalent in the Isabelle/HOL proof assistant [32, 33]. This ensures that the language qualifies for two key properties:

compositionality The semantic composition of two models yields the semantics of the supermodel. This is notably emphasized by the property of *stuttering-invariance* which shows that the addition of observation instants does not “break” a run and preserves specification satisfiability.

executability The specification is constructive and allows the derivation of execution traces. This allows for trace generation for testing and simulation purposes.

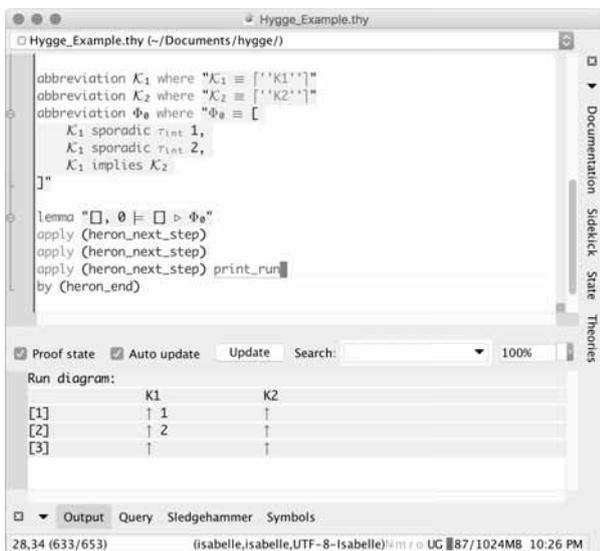


Figure 7: Executing the operational semantics in the Isabelle/HOL proof assistant

6 Related Work

The aim of our study and its purpose to the aviation community is similar to the goal of the real-time on-board Fault Detection and Diagnosis introduced by Goupil *et al.* [34] but differs in the scope of their study. Their study focuses on numerical aspects of signals and emphasizes on the link between academic and industrial R&D. Our research work aims at bringing new knowledge regarding timed aspects of discrete-event models.

Chhaya *et al.* [35] introduced the Aviation Scenario Definition Language (ASDL) which provides a

Domain-Specific Language for scenarios of the same kind of our study. They provide an extensive framework to design aviation-related procedures. On the mathematical side, specifications consider only logical time and are translated into finite-state machines which employ traditional model-checking for verification purposes. Compared to our approach, our language aims at remaining a general-purpose and multi-level coordination framework, combined with the ability of specifying constraints containing chronometric time and physical quantities.

On a system-oriented level, Lustre/SCADE [36] is a well-known asset for the development of critical embedded systems. It has been qualified for DO-178B and is used for specifying flight control systems onboard the Airbus A340-600 and A380 [37]. Lustre/SCADE considers a unique and global driving clock for all specified components, whereas our framework allows to specify independent time islands where no global clock may exist. Moreover, time in SCADE is purely logical and does not consider chronometric time. In the problem we attempt to address, it may be necessary to consider the latter while designing and specifying real-time systems. Indeed, time delay constraints are crucial for closed loop controls of FCSs as time lags may exist due to computing, latency or storage [37].

Dealing with test oracles [38], our framework rightfully determines correctness on the outputs of a system. Indeed, the previously mentioned semantics allows to exhaustively generate execution traces that are correct with respect to specifications. The test oracle consists in filtering out generated branches which no longer satisfy the current outputs. The test oracle detects a violation whenever all possibly-satisfying branches have been filtered out.

7 Future Work

The permissive nature of the language allows to leave time relations between different quantities unspecified by default. Time related clocks can be gathered and are said to live in *time islands*. Leaving them unrelated means that they live independently. This feature is particularly interesting for *distributed computing* as it is not always possible to determine how time flows in one computing unit relatively to another. Yet, it has to be made sure that any computation is completed eventually. A similar synchronization mechanism can be found, for instance, in the Airbus A380 [39].

8 Conclusion

Our study highlights the TESL specification language as a unified environment for modeling and validation along the different stages of (1) designing models, (2) running their simulation, and (3) monitoring their runtime-compliance. We have presented a case study of fundamental operational scenarios found in the aviation industry, with high-level models addressing large-scale systems. In particular, our language and its associated high-level specifications are agnostic to concrete hardware implementations, providing a suitable framework for testing and monitoring systems similarly to black-box testing. We believe our framework also addresses the current trend for distributed computing [40] which is increasingly finding its way in critical embedded systems.

References

- [1] Boulanger F, Jacquet C, Hardebolle C, Prodan I. TESL: a Language for Reconciling Heterogeneous Execution Traces. In: *Formal Methods and Models for Codesign (MEMOCODE), 2014 Twelfth ACM/IEEE International Conference on*. Lausanne, Switzerland. 2014; pp. 114–123.
URL <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6961849>
- [2] Liu X, Liu J, Eker J, Lee EA. *Heterogeneous Modeling and Design of Control Systems*, chap. 7, pp. 105–122. Wiley-Blackwell. 2005;.
URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/047172288X.ch7>
- [3] Ptolemaeus C, ed. *System Design, Modeling, and Simulation using Ptolemy II*. Ptolemy.org. 2014.
URL <http://ptolemy.org/books/Systems>
- [4] Eker J, Janneck JW, Lee EA, Liu J, Liu X, Ludvig J, Neuendorffer S, Sachs S, Xiong Y. Taming heterogeneity - the Ptolemy approach. *Proceedings of the IEEE*. 2003;91(1):127–144.
- [5] Boulanger F, Hardebolle C, Jacquet C, Marcadet D. Semantic Adaptation for Models of Computation. In: *2011 Eleventh International Conference on Application of Concurrency to System Design*. 2011; pp. 153–162.
- [6] Vara Larsen ME, Deantoni J, Combemale B, Mallet F. A Behavioral Coordination Operator Language (BCoL). In: *18th International Conference on Model Driven Engineering Languages and Systems (MODELS 2015)*. 2015; .
- [7] Pnueli A. The temporal logic of programs. In: *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*. 1977; pp. 46–57.
- [8] *Annex 8 - Airworthiness of Aircraft*. International Civil Aviation Organization. 2018.
URL <https://store.icao.int/products/annex-8-airworthiness-of-aircraft>
- [9] *Easy Access Rules for Air Operations (Regulation (EU) No 965/2012)*. European Union Aviation Safety Agency. 2019.
URL <https://www.easa.europa.eu/document-library/general-publications/easy-access-rules-air-operations>
- [10] Ferrell TK, Ferrell UD. RTCA DO-178C/EUROCAE ED-12C. In: *Digital Avionics Handbook*, pp. 16–1. Taylor & Francis. 2017;.
- [11] RTCA DO-178C. *Software Considerations in Airborne Systems and Equipment Certification*. RTCA Inc., Washington, D.C. 1992. This document is also known as EUROCAE ED-12C in Europe.
- [12] Traverse P. Airbus Electrical Flight Controls: A Family of Fault-Tolerant Systems. In: *Digital Avionics Handbook*, pp. 31–1. Taylor & Francis. 2015;.
- [13] Spitzer C, Ferrell U, Ferrell T, eds. *Digital Avionics Handbook*. Electrical engineering handbook series. Boca Raton: CRC Press. 2015.
- [14] Weibel RE, Hansman RJ. Safety considerations for operation of unmanned aerial vehicles in the national airspace system. *Tech. rep*. 2006.
- [15] Garcia R, Barnes L. Multi-UAV Simulator Utilizing X-Plane. *Journal of Intelligent and Robotic Systems*. 2009;57(1):393.
URL <https://doi.org/10.1007/s10846-009-9372-4>
- [16] Jung D, Tsiotras P. Modeling and hardware-in-the-loop simulation for a small unmanned aerial vehicle. In: *AIAA Infotech@ Aerospace 2007 Conference and Exhibit*. 2007; p. 2768.
- [17] Livadas C, Lygeros J, Lynch NA. High-level modeling and analysis of the traffic alert and collision avoidance system (TCAS). *Proceedings of the IEEE*. 2000; 88(7):926–948.
- [18] Mallet F. Clock constraint specification language: specifying clock constraints with UML/Marte. *Innovations in Systems and Software Engineering*. 2008;4(3):309–314.
- [19] André C. Syntax and Semantics of the Clock Constraint Specification Language (CCSL). *Research Report*

- RR-6925, INRIA. 2009.
URL
<https://hal.inria.fr/inria-00384077>
- [20] Lee EA, Sangiovanni-Vincentelli AL. The Tagged Signal Model A Preliminary Version of a Denotational Framework for Comparing Models of Computation. *Tech. Rep. UCB/ERL M96/33*, EECS Department, University of California, Berkeley. 1996.
URL <http://www2.eecs.berkeley.edu/Pubs/TechRpts/1996/3030.html>
- [21] Cessna Aircraft Company. *Cessna Skyhawk Information Manual*.
URL <http://www2.eecs.berkeley.edu/Pubs/TechRpts/1996/3030.html>
- [22] Linton JO. The physics of flight: I. Fixed and rotating wings. *Physics Education*. 2007;42(4):351.
URL <http://stacks.iop.org/0031-9120/42/i=4/a=003>
- [23] Airbus. *Airbus A320 - Front Panel*. 2007.
- [24] Airbus. *A318/A319/A320/A321 Flight Crew Training Manual*. 2002.
- [25] Ladkin PB. Analysis of a technical description of the Airbus A320 braking system. *High Integrity Systems*. 1995;1:331–350.
- [26] Airbus. *A318/A319/A320/A321 Performance Training Manual*. 2006.
- [27] Nguyen Van H, Balabonski T, Boulanger F, Keller C, Valiron B, Wolff B. A Symbolic Operational Semantics for TESL - With an Application to Heterogeneous System Testing. In: *Formal Modeling and Analysis of Timed Systems - 15th International Conference, FORMATS 2017, Berlin, Germany, September 5-7, 2017, Proceedings*. 2017; pp. 318–334.
- [28] Milner R, Tofte M, Macqueen D. *The Definition of Standard ML*. Cambridge, MA, USA: MIT Press. 1997.
- [29] Westrick S, Yadav R, Fluet M, Acar UA. Disentanglement in Nested-Parallel Programs. *Proc ACM Program Lang*. 2019;4(POPL).
URL <https://doi.org/10.1145/3371115>
- [30] Weeks S. Whole-program Compilation in MLton. In: *Proceedings of the 2006 Workshop on ML, ML '06*. New York, NY, USA: ACM. 2006; pp. 1–1.
URL <http://doi.acm.org/10.1145/1159876.1159877>
- [31] Nguyen Van H, Boulanger F, Wolff B. A Formal Development of a Polychronous Polytimed Coordination Language. *Archive of Formal Proofs*. 2019;2019.
URL https://www.isa-afp.org/entries/TESL_Language.html
- [32] Nipkow T, Klein G. *Concrete Semantics: With Isabelle/HOL*. Springer Publishing Company, Incorporated. 2014.
- [33] Nipkow T, Wenzel M, Paulson LC. *Isabelle/HOL: A Proof Assistant for Higher-order Logic*. Berlin, Heidelberg: Springer-Verlag. 2002.
- [34] Goupil P, Dayre R, Brot P. From theory to flight tests: Airbus Flight Control System TRL5 achievements. *IFAC Proceedings Volumes*. 2014;47(3):10562 – 10567. 19th IFAC World Congress.
URL <http://www.sciencedirect.com/science/article/pii/S147466701643291X>
- [35] Chhaya B, Jafer S, Durak U. Formal Verification of Simulation Scenarios in Aviation Scenario Definition Language (ASDL). *Aerospace*. 2018;5(1).
URL
<https://www.mdpi.com/2226-4310/5/1/10>
- [36] Halbwachs N, Caspi P, Raymond P, Pilaud D. The synchronous data flow programming language LUSTRE. *Proceedings of the IEEE*. 1991; 79(9):1305–1320.
- [37] Ameer YA, Boniol F, Wiels V. Toward a wider use of formal methods for aerospace systems design and verification. *International Journal on Software Tools for Technology Transfer*. 2009;12:1–7.
URL <https://doi.org/10.1007/s10009-009-0131-4>
- [38] Durrieu G, Waeselynck H, Wiels V. LETO - A Lustre-Based Test Oracle for Airbus Critical Systems. In: *Formal Methods for Industrial Critical Systems*, edited by Cofer D, Fantechi A. Berlin, Heidelberg: Springer Berlin Heidelberg. 2009; pp. 7–22.
- [39] Sánchez-Puebla MA, Carretero J. A new approach for distributed computing in avionics systems. In: *Proceedings of the 1st international symposium on Information and communication technologies*. Trinity College Dublin. 2003; pp. 579–584.
- [40] Hildenbrand Y. ED-247 (VISTAS) Gateway for Hybrid Test Systems. In: *SAE Technical Paper*. SAE International. 2018; .
URL
<https://doi.org/10.4271/2018-01-1949>

Examining Repositories for Simulation Data

Oliver Ullrich^{1*}, Victor Potapenko², Naphtali Rishé²

¹Fraunhofer Institute for Intelligent Analysis and Information Systems IAIS, Schloss Birlinghoven, 53757 Sankt Augustin, Germany, *oliver.ullrich@iais.fraunhofer.de

²School of Computing and Information Sciences, Florida International University, Miami, FL 33199, USA

SNE 30(2), 2020, 61 - 66, DOI: 10.11128/sne.30.tn.10513
 Received: February 26, 2020; Revised May 21, 2020;
 Accepted: May 25, 2020
 SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
 ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Researchers and practitioners often struggle finding or generating adequate data to design, calibrate, or validate simulation models. This leads to greater time and effort allocated to searching for or producing data, rather than performing scientific research itself. This data barrier is especially cumbersome in the long tail of computer science – smaller laboratories typically without access to larger institutions' data sources.

This review examines nineteen existing data repositories based on their feature sets. Out of these reviewed systems, only six have advanced feature sets that are significantly different from standard digital libraries. No single data repository provides a combination of features and tools geared towards simulation projects conducted at smaller laboratories, and none offers features that would allow for purchase or sale of data.

Introduction

Research projects to advance modelling and simulation methods often depend on the availability of suitable and reliable data to design, calibrate, and validate models. Significant time and effort is spent on finding or collecting, assessing, licensing, and pre-processing such data sets even before the actual modelling and simulation project can begin. This data barrier is especially cumbersome in the long tail of computer science – smaller laboratories typically without access to larger institutions' data sources.

This review examines nineteen existing data repositories (see Figure 1) based on their feature sets and utility to both data owners and data users, and with specific consideration of the long tail of computer science.

These results could also be seen as a call to action: There is definitely room for a data repository and arbiter platform, incorporating functionality aimed at supporting computer scientists in the long tail of the field, promising honorary and even financial motivation for data owners to curate and share their research data.

The paper continues with sharing some background on data repositories (see Section 1), followed by a discussion of desirable functionality, and specifically on features desired by both data owners and/or data users (see Section 2). The paper then goes on to discuss the 19 examined platforms and takes a closer look at six of the most promising of them (see Section 3). It closes with a short summary of the lessons learned and an outlook on further research (see Section 4).

1 Background

A *data repository* is a shared data storage resource that holds multiple types of data to be used for analytical or modelling purposes (see [1]), providing users at least with means to upload, manage, search, and download data sets. Some of these platforms provide more advanced functions that might include tagging, querying, versioning, and code integration (see Figure 1).

Studies (see [16] and [1]) focused on data providers across a variety of fields demonstrate a market in a stable and highly innovative phase that is still being dominated by a high vertical integration with lack of intermediaries indicating limited market efficiency. Similar conditions are apparent in the long tail of computer science, where scientists who have no particular incentive or specialized platform to share their data with the rest of the scientific community make most discoveries in a large number of smaller, silo-like laboratories.

The current trends point towards domain-focused, self-generated, specialized data (see [16]). These trends are well aligned with needs of the computer science community.

There is a need to create an ecosystem that allows its participants to organize into communities, create, curate, and interlink their own sub-repositories, integrate data with code, trace data and code evolution via dataset versioning, publish and subscribe to near real-time data feeds, access the data via industry standard APIs, effectively manage licensing, and trade the data based on its value (see [16] and [1]). At the time of writing, it is estimated (see [17] and [1]) that providers of data in the scientific domain offer it cost free approximately 80% of the time, so that methods and tools to widely share data for honorary purposes, for example acknowledgements or co-authorships, are needed as well.

2 Potential Services

Consider this scenario: A team has developed an idea, a software tool or a new simulation technique or model to be tested. But how can the team get data to calibrate and validate these models? Where can large datasets required to evaluate the software tool be found? Once acquired, can the whole data set be included in a publication?

To solve these issues a platform or service would be needed aimed specifically at the needs of computer science researchers, motivating *data owners* to further disseminate already existing, valuable data. Such a system would enable them to monetize data sets and/or to get proper honorary acknowledgment to the data producers and their sponsoring agencies and programs, as well as to benefit from references to research papers and patents resulting from data access.

On the *data users'* side, all computer scientists are potential users of such a repository system. Especially smaller research groups without access to large data producing facilities would gain access to curated, diverse, vast amounts of data.

In the next few paragraphs some advanced functionality beyond simple data up- and downloads is envisioned that would facilitate data sharing and availability.

2.1 Services to Data Owners

An ideal system would provide data owners with a web based, encryption-enabled interface allowing researchers to deposit domain-specific datasets, to create citable Digital Object Identifiers (DOI), to define sample datasets, and to store relevant meta-data on the dataset and its owner, available licenses, pricing, when relevant, and the dataset's range, quality, and domain.

A step further, such a system would offer license brokering and management to data owners and data consumers, including an easy to use, visual expert system ("wizard") that helps data owners to find the best possible license model custom-tailored to their needs and wishes. A dataset owner could then determine in which way the dataset can be used: for research, commercial or non-commercial use, whether anonymization or pseudonymization is required, whether only summaries can be published and in what aggregation, what fees and what terms of non-disclosure apply.

To further facilitate data reuse, such a system would enable data owners to enforce fee collection for various license types, for commercial or non-commercial use. This way of monetizing available data entices further collaboration and data sharing between research groups.

For commonly used file and stream formats, the platform could offer value-added services to data owners that enrich datasets and simplify data preparation, including auto-anonymization or pseudonymization, geo-coding, geo-tagging, visualization, auto-aggregation, and the (semi-) automatic generation of sample data.

Such a system would include, for each deposited dataset, acknowledgments to sponsoring agencies and programs, and references to research papers and patents resulting from data access. The system would generate reports on how many papers, patents, projects, and other artifacts result from data access, sorted by dataset, owner, or sponsoring agency or program. In combination with the rating of datasets and transactions by both owners and users, this reporting generates a certain degree of peer pressure, in addition to the formality of the license terms, to ensure full and proper acknowledgment.

2.2 Services to Data Users

With the envisioned platform, interested researchers would be able to browse deposited datasets by category, domain, license, collection and deposition date, and other attributes, download sample data, and check available licenses. For common file and stream formats, the system would offer data previews in a web browser, including table-based views, aggregations, simple statistics, and visualization. If questions remain, the platform allows the interested data consumer to contact the dataset owner. It is feasible that a repository system could interface, should the collection of a fee be required, to secure external payment services to facilitate the transaction.

Once a deal is struck, the user can download it or use the system's API to access the dataset.

To encourage data sharing and cooperative behavior, a system might offer trust-building community functions, including the rating of datasets and transactions, as well as integration with research social networks. Using gamification measures, users could be encouraged to review datasets and to rate them according to their quality and range. In addition, the platform could help to build communities of users curating groups of datasets, and to discuss their strengths and weaknesses.

Such a system would offer advanced functions for data access, programmed searches, and allow data owners to dynamically update datasets once new data becomes available – data consumers' listeners registered with corresponding datasets would be automatically notified as soon as any updates take place. The system would also allow code and data integration, would provide an interface to and manage references to Git code repositories. Optionally, the data resulting from these algorithms could in turn be stored in the system via its API, thereby adding value to already present data.

The envisioned system would be open to all computer science researcher, regardless of specialization and research field, whether working in industry or in academia. While other repositories focus on bringing together specialized data based on distinct fields of origin (e.g. geology, genetics, or marine biology), the platform would focus on data supporting the computing research data consumption needs, regardless of the domain of data origin. It would therefore especially suit the needs of modelling and simulation researchers.

Figure 1 depicts a distillation of the envisioned services.

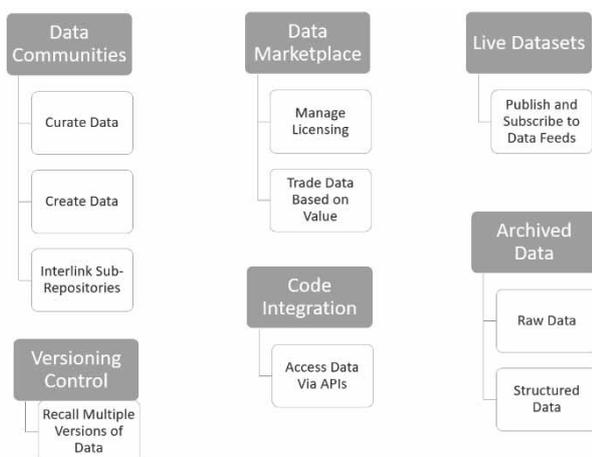


Figure 1: A summary of envisioned functionality.

3 Repositories for Simulation Data

3.1 Overview

The multitude of currently existing data repositories address widely varied data requirements of commercial and non-profit institutions, as well as the individual researchers in the field of computer science. While some provide data directly relating to the field of computer science, such as machine learning data sets, data encryption, or operating systems, others provide multidisciplinary datasets, but are geared and tooled specifically towards users that come from a computer science background and require advanced features.

An overwhelming majority of these repositories are designed to mimic digital libraries, which have data storage and retrieval features, might support basic versioning, and contain multidisciplinary datasets. These digital libraries have basic functionality and do not support advanced data retrieval via an API, or the interlinking of related datasets. Out of the 19 repositories examined as a sample (see Figure 2), only six of the systems reviewed that provide more advanced functionality supporting the requirements of computer scientists, more specifically to simulation modellers (see Figure 3).

These repositories with advanced functionality beyond that of a standard digital library are Figshare [8], Zenodo [23], Unidata Internet Data Distribution [22], CKAN @ IoT Lab [5], CITK [4], and GitHub [9].

3.2 A Closer Look

The following six dataset repositories contain features especially geared towards supporting computer science research (see [17] and [16]):

Figshare. Figshare is general-purpose cross-disciplinary data repository based in Great Britain. It is one of the most popular repositories and houses more than 500,000 datasets, which is more than any other data repository surveyed. It has a seamless intuitive user interface and allows for storage, sharing, interlinking, and discovery of a multitude of artifacts, including figures, media, datasets, file-sets, posters, papers, presentations, thesis, and code. Figshare supports access via a REST API as well as the industry standard OAI-PMH.

User workspace offered by Figshare allows users to create and manage projects by adding or removing artifacts related to it. It has facilities to link collections to

the project and observe the project activity over time. The workspace is another key differentiating feature that motivates the data publishers and consumers to work within the service's ecosystem and perpetuate it by storing and sharing the results of their research based on data gathered from Figshare through the service itself.

In-browser data preview is a data visualization feature of Figshare. It allows for data of various types to be visualized instantly within browsers to provide the user with a general sense of the data before the user initiates a download. This functionality reduces the stress on the overall system by helping to prevent the user from downloading unneeded datasets.

Figshare is missing the support for live updates to datasets, data versioning mechanisms, community-based data curation, and only allows to link code repositories from GitHub. The service does not host code repositories itself, which limits the level of integration between data and code repositories. Furthermore, it does not provide means to trade data.

Zenodo. Zenodo is an open data catch-all cross-disciplinary repository for research funded by the European Commission, also supported and funded by CERN. Similarly to Figshare, the service provides access via REST and OAI-PMH. Each upload gets its own Digital Object Identifier (DOI), which makes it uniquely identifiable and citable. The repository accepts various data types, including publications, datasets, software, and presentation. It also allows for in-browser data visualization, which enables users to judge the fit of data for a particular purpose prior to download initiation. The service provides the facilities to identify grants used in research as well as flexible licensing that allows for sharing of datasets among communities.

Zenodo's approach to the user workspace within a data repository service is somewhat different from that of Figshare. The service is organized around the concept of communities. Consumers and publishers are encouraged to organize into communities, where they can create their own sub-repositories and curate the data that gets deposited. It means that Zenodo has the facilities to self-organize into meaningful groups that work on similar goals and datasets as well as self-curate to ensure that data is relevant, valid, and generally useful.

Name	Key Functionality	Reference
BABS	Basic digital library for the humanities	[2]
CCITK	Storage of data, data processing code, and derived data sets	[4]
CIESIN	Basic digital library for earth science data	[3]
CKAN	Storage of data, data processing code, and derived data sets	[5]
Clarin	Basic digital library for language resources	[6]
Dataverse	Repository system for basic digital libraries	[7]
Figshare	User orientation, in-browser visualization, data versioning	[8]
GEON	Basic digital library	[10]
GitHub	Storage for code and accompanying data, versioning	[9]
NatureServe	Digital repository for biodiversity data, provides API for access	[11]
OLAC	Basic digital library for language resources	[12]
Pachyderm	Platform to host digital libraries with additional versioning	[13]
PredictDB	Digital library for genome data and prediction models	[14]
RAMADDA	Digital library for satellite data	[15]
SNAP	Repository for network-related data from several disciplines	[18]
UA-CR	Basic digital library	[19]
UCI	Basic digital library for large standardized data sets to test machine learning algorithms	[20][21]
Unidata	Digital library plus real-time delivery	[22]
Zenodo	Community orientation, in-browser visualization, data versioning, high searchability	[23]

Figure 2: A sample of 19 data repository systems was examined.

The service is implemented using Invenio, a free open-source digital library framework originally developed at CERN. The framework covers all aspects of digital library management and allows for diverse content types, including articles, books, journals, photos, videos, and datasets. It provides digital library features, including navigable collection tree, powerful search engine, flexible metadata, collaborative features, and personalization.

With this type of framework, it is relatively inexpensive to get a basic digital library-type data repository running. Consequently, a software engineering team may quickly shift focus towards integration of more advanced functionality.

Much like Figshare, Zenodo is missing the means to trade data, live dataset updates, data versioning mechanisms, and data market. Unlike Figshare, it does provide the support for content curated via communities.

Unidata. Unidata Internet Data Distribution (UIDD) is a community of 260 universities sharing tools to disseminate near real-time earth observation data online. While offering the standard data storage, retrieval and discovery features, this service is designed to automatically deliver certain datasets to subscribers as soon as the data becomes available. In other words, a publisher of data can establish a link to the repository to deposit data from the publisher's sensors in real-time. Subsequently, a consumer of data can subscribe to the published data feed and receive dataset updates in near real-time. This concept of "live datasets" is appealing as it makes the datasets dynamic, which enables not only research opportunities based on most up-to-date data, but also to some extent the creation of applications that showcase whether or not the research findings remain valid when provided with new data points, which were not in the original data set.

Unidata is missing all of the advanced functionality of Figshare and Zenodo. The only functionality that sets it apart from a digital library is "live datasets", which is surprisingly absent from all other repositories with advanced functionality.

CKAN @ IoT Lab and **CITK** are both data repository platforms that focus on research data and software code integration. These repositories position themselves as toolkits for researchers, because they implement facilities to store and manage datasets as well as software code associated with datasets.

This type of functionality is beneficial for researchers in the computer science field because many researchers use custom-built software to perform research using external datasets, and it only makes sense to be able to store the code alongside the corresponding data set. Furthermore, it makes sense to store the resulting dataset as a derivative of the original and interlink the original dataset, software code, and result dataset within the same repository. This idea builds further on traceability of the evolution of datasets and significantly improves the motivational aspect for the data consumers to store and share their work within the repository ecosystem.

CKAN also functions as an open source data portal. The implemented features allow to publish and find datasets, store and manage data, and engage with users. It is also highly extendable and customizable, has advanced geospatial and visualization features, and includes a RESTful JSON API for querying and accessing the dataset information. This software can be used to create a data repository with basic features quickly and build extensions necessary to support more advanced features tailored to the computer science community.

CKAN and CITK strive to provide data and code integration, however, they are missing dataset versioning functionality, which is key to building a useful research collaboration and data trading platform. There is no support for communities, workspaces, or commercial data exchange markets.

GitHub. At the opposite end of the dataset/code spectrum is the widely-used GitHub code repository. The repository is built using Git open-source software, which is created to share, track, manage and execute simple and complex software projects. Git is one of the most widely used team software code management technologies by computer scientists around the world. It allows to create a code repository, share it, and collaborate on it while mitigating conflicts between the changes made to the code by the participants in the sharing process. It also allows for the creation of derivations of the source code, thereby enabling project evolution, while maintaining full traceability of changes made by all participants. GitHub is not designed for dataset storage, the underlying Git technology was created with source code in mind. However, the overall code storage, sharing, management, evolution, and traceability principles are applicable to pure dataset repository realm.

Since code is essentially data, a data repository geared towards researchers within the computer science field should provide the facilities to deposit and manage code alongside the relevant datasets.

GitHub does not have the facilities to store large amounts of data. Git technology is built for line-by-line code versioning over a large number of individual files, and is not applicable for data repository purposes.

Figure 3 depicts an overview of the main function groups aimed at supporting computer scientists that the six discussed data repositories feature.

Feature	CITK	CKAN	Figshare	GitHub	Unidata	Zenodo
Communities	x	x	x	x	x	✓
Marketplace	x	x	x	x	x	x
Code Integration	✓	✓	✓	✓	✓	✓
Versioning	x	x	x	✓	x	x
Live Datasets	x	x	x	x	✓	x

Figure 3: An overview of the main feature groups of selected data repository platforms aimed at supporting computer scientists.

4 Conclusion

This paper presented an overview of 19 data repository systems in the area of computer science. Out of these reviewed systems, only six have advanced feature sets that go significantly beyond standard digital libraries. No single data repository provides a combination of features and tools geared towards simulation projects conducted at smaller laboratories, and none offers features that would allow for purchase or sale of data.

Among other considerations, the existing platforms are especially failing to create a marketplace environment where computer scientists are enticed to share their own data, evaluate and provide feedback on the data submitted by others, and pay a fair price for licensing rights to the peer-reviewed data. Such a platform would enable market participants to add value to original datasets by creating scripts that derive versions of originals, which can be used for further, non-obvious modeling and analysis, while appropriately crediting the original dataset.

References

- [1] Assante M, Candela L, Castelli D, Tani A. Are Scientific Data Repositories Coping with Research Data Publish-

ing? *Data Science Journal*, vol. 15, 2016.

- [2] “Bayerischen Staatsbibliothek und Staatsarchive, Langzeitarchivierung”, <https://www.digitale-sammlungen.de/index.html>, accessed 17-Feb-2020.
- [3] “Center for International Earth Science Information Network”, <http://www.ciesin.org/data.html>, accessed 17-Feb-2020.
- [4] “Cognitive Interaction Toolkit”, <https://toolkit.cit-ec.uni-bielefeld.de>, accessed 7-Jan-2017.
- [5] “CKAN @ IoT Lab”, <http://ckan.iotlab.eu>, accessed 7-Jan-2017.
- [6] “CLARIN”. <https://www.clarin.eu/>, accessed 17-Feb-2020.
- [7] “Dataverse”, <https://dataverse.org/>, accessed 17-Feb-2020.
- [8] “figshare”, <https://figshare.com>, accessed: 7-Jan-2017.
- [9] “GitHub”, <https://github.com>, accessed 7-Jan-2017.
- [10] “GEONGRID – Global Earth Observation Network”, <http://geongrid.org>, accessed 16-Dec-2018.
- [11] “NatureServe Explorer”, <http://explorer.natureserve.org/>, accessed 18-Feb-2020.
- [12] “OLAC – Open Language Archives Community”, <http://www.language-archives.org/>, accessed 17-Feb-2020.
- [13] “Pachyderm”, <https://www.pachyderm.com/>, accessed 17-Feb-2020.
- [14] “PredictDB Data Repository”, <http://predictdb.org/>, accessed 18-Feb-2020.
- [15] “RAMADDA on the NSF Jetstream Cloud”, <https://ramadda.scigw.unidata.ucar.edu>, accessed 18-Feb-2020.
- [16] Schomm F, Stahl F, and Vossen G. Marketplaces for data. *ACM SIGMOD Record*, vol. 42, no. 1, p. 15, 2013.
- [17] Stahl F, Schomm F, Vossen G, and Vomfell L. A classification framework for data marketplaces. *Vietnam Journal of Computer Science*, vol. 3, no. 3, pp. 137–143, 2016.
- [18] “SNAP – Stanford Large Network Dataset Collection”, <https://snap.stanford.edu/data/>, accessed 18-Feb-2020.
- [19] “University of Arizona – Campus Repository”, <https://repository.arizona.edu/arizona/>, accessed 18-Feb-2020.
- [20] “UCI Knowledge Discovery in Databases Archive”, <http://kdd.ics.uci.edu/>, accessed 18-Feb-2020.
- [21] “UCI Repository”, <http://ics.uci.edu/~mlearn/MLRepository.html>, accessed 7-May-2017.
- [22] “Unidata” Internet Data Distribution, <http://www.unidata.ucar.edu/projects/index.html#idd>, accessed 7-Jan-2017.
- [23] “Zenodo – Research. Shared.”, <https://www.zenodo.org>, accessed 7-Jan-2017.

Simulation in Wood Science: using the Process Gluing as an Example

Carina Rößler^{1,2*}, Martin Riegler¹

¹ Wood K plus, Competence Centre for Wood Composites and Wood Chemistry, Area Wood Materials Technologies, Konrad-Lorenz-Strasse 24, 3430 Tulln, Austria; **carina.poell@asak.at*

² Institute of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria

SNE 30(2), 2020, 67 - 72, DOI: 10.11128/sne.30.tn.10514
Received: July 20, 2019 (Selected ASIM SST 2018 Postconf. Publ.), Accepted: December 15, 2019
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. In this paper, a mathematical model for the sub-process of gluing in the manufacturing process of particleboards is presented. The model describes the application of resin droplets onto wood particles on laboratory scale. At the moment the model and simulations are carried out for the problem considered in two dimensions. The developed mathematical model uses lattice gas cellular automata for the movement of the wood particles, and random walk for the movement of the resin droplets. Feasibility was shown with simulation results of a greatly simplified mathematical model in two dimensions. This model was developed further for a better fitting to the real process. First simulation results of this more detailed model are presented. The next step will be the specification of how validation will be performed. In future, it is planned to create a mathematical model in three dimensions.

Introduction

The industrial manufacturing process of wood-based composites consists of several consecutive sub-processes. The quality of the final product depends on the raw material and the different processing steps. To ensure a high quality of the product, while optimizing the use of energy and resources, adjustable process parameters need to be continuously adapted [1]. Simulations of potential measures can predict the impact of these measures and hence avoid possible costs of rejects.

This generates a deeper understanding of the manufacturing process and enables a more efficient usage of resources.

Based on a value benefit analysis, the sub-process gluing was selected for modelling and simulation.

The wood particles are located in a mixer and thus are moved. Due to the position in the mixer the particles have a local boundary condition. Their movement is locally limited (chips are located at the bottom of the mixer), unless they are transported upwards from the mixer. The resin is sprayed from above into the mixer using nozzles. The resin drops hit the moving wood particles. The aim is to describe the movement and interactions of the wood and resin particles, which results in computing the resin distribution. Appropriate geometries for resin particles and wood particles needed to be selected and different size distributions to be applied. The rules for the different kinds of collisions needed to be defined. In the case of a collision between wood and resin, a statistical distribution was used for calculating the quantity of resin, which penetrates, adheres and passes to another wood particle (lubrication effect).

1 Gluing

In this section, the process of gluing is described according to [2], [3], [4] and [5]. Gluing is a sub-process of the manufacturing process of particleboards. In this process step the wood particles are glued with the resin formulation, which is sprayed into the mixer using nozzles. Based on [2] the typical dimensions of the wood chips are: length 8-10 mm, width 1.5-3.5 mm and thickness 0.25-0.4 mm. In [3] it is stated that the diameter of the resin droplets is between 30 μm and 100 μm .

The goal of gluing is to equally distribute the resin droplets on the surface of the wood particles, which is important for the bonding of the wood particles. This has a strong impact on the properties of the final particleboard. In the following, the process of gluing is considered on laboratory scale by using a so-called ploughshare mixer, which is shown in Figure 1.



Figure 1: Picture of a ploughshare mixer.

Within the mixer used on laboratory scale, there are four ploughshares (also called mixing arms), a milling head, and an opening for the nozzle. First, the wood chips are located at the bottom of the mixer. Due to the mixing arms the wood particles are moved within the mixer, which constitutes a local boundary condition. The resin is sprayed into the mixer using a nozzle. The resin droplets hit the moving wood particles and thus the wood particles are glued.

2 Model

At the moment the model is developed for the two dimensional problem. For modelling the movement of the wood chips a lattice gas cellular automaton (LGCA) is used. The common LGCA is described in [6]. For using LGCA in this setting, the method has to be extended and modified, which will be explained later. For the LGCA a lattice, states and the evolution ε , where \mathcal{C} is the collision operator and S is the streaming operator,

$$\varepsilon = S \circ \mathcal{C}$$

have to be defined. The movement of the resin droplets is realised by a random walk across the lattice used in the LGCA.

The elements, which are used within this model, and the properties of the environment are described in the following.

- wood particles: The shape of the wood particles is considered to be a rectangle.
- resin particles: The shape of the resin droplets is considered to be a circle.
- resinating mixer: The geometry of the cross-sectional area of the resinating mixer is assumed to be a circle.
- ploughshares/mixing arms: The ploughshares move along a circle line.
- time interval
- grid width

In the following, the main elements of the LGCA are considered in more detail. In [7], which is a previous publication of the model, these are already described.

2.1 Lattice

The lattice is built by nodes, which are the vertices of equilateral triangles. Therefore, the lattice shows hexagonal symmetry. For obtaining “smooth” objects (circle, which builds the resinating mixer; rectangles, which build the wood particles; circles, which build the resin droplets), a suitable grid width has to be chosen.

In Figure 2 above, the black points are the vertices of the equilateral triangles, which are the nodes of the lattice. The red dashed line represents the wall of the resinating mixer. In Figure 2 below, a small part of the cross-sectional area of the mixer is depicted. The black lines are the edges of the equilateral triangles and the smaller colored triangles represent the directions of movement.

According to [6] at each node there are six cells, which represent the six directions of movement. These correspond to the side lengths of the equilateral triangles and are the so-called lattice velocities. These common lattice velocities are given in equation (2) for $i = 1, \dots, 6$.

$$c_i = \left(\cos\left(\frac{\pi}{3}i + \frac{\pi}{2}\right), \sin\left(\frac{\pi}{3}i + \frac{\pi}{2}\right) \right)$$

On the one hand, wood particles are stationary, if they are not moved by a mixing arm. On the other hand, resin particles can be stationary, if they stick to the wall of the resonating mixer. Due to the fact that particles can have velocity zero, the common LGCA has to be extended regarding the possible lattice velocities. Therefore, at each node a seventh cell is added, whereby its lattice velocity is zero.

Thus, the lattice velocity of this cell is defined by equation (3).

$$c_0 = (0,0)$$

Due to the different scale of the wood and resin particle, and the corresponding grid width, the wood particles extend over several nodes of the lattice, which is not provided within the common LGCA. Therefore, some specifications regarding the states, collisions and streaming are necessary.

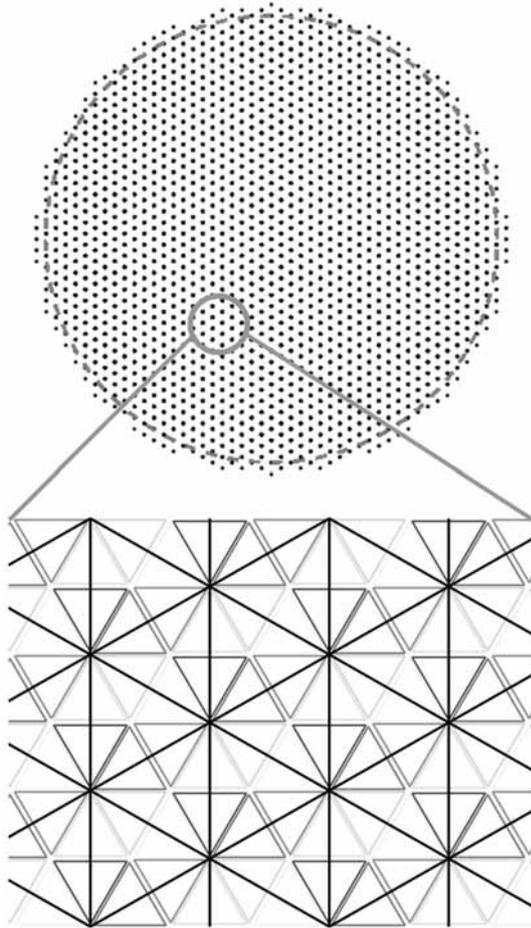


Figure 2: Schematic depiction of the lattice points and the lattice velocities.

2.2 State

Each cell of a node must have a so-called state. The state indicates what kind of particle or part of the resinating mixer is located at a certain position. There are certain states, which are described in the following.

- empty: The state “empty” is used for the air, which will not be modelled, and for the definition of the state of a node, which will be explained later.
- wall: The state “wall” is used for the rigid body of the ploughshare mixer.
- ploughshare/mixing arm: For the moving mixing arms this state is used.
- milling head: The state used for the milling head.
- wood: The state for parts of a wood particle without resin.
- resin: The state for the resin droplets.
- glued wood: The state for parts of a wood particle, where resin is adhered, i.e. the state wood is changed to glued wood, if a wood particle and a resin particle collide with each other. After such a collision the resin droplet penetrates into the wood particle over time. Furthermore, if a glued wood particle collides with an unglued wood particle, part of the resin droplet passes to the unglued wood particle, which is called lubrication effect. In section 2.3 these behaviors will be included in the collision rules.

Corresponding to the dimensions of the wood particle and the grid width, the wood particle expands over several nodes. For each node, which is occupied by the wood particle, the state of the cell, which corresponds to the direction of movement, is set to wood. Due to the fact, that each node, which is occupied by the same wood particle, has to move in the same direction. For the other cells of the nodes, the states are set to empty. According to the described situation the cells, which are set empty, of a node, which belongs to a wood particle, are also occupied by the wood particle. Thus, a modification of the common LGCA is necessary. Therefore, the state of a node is introduced. It is defined as wood, if one cell at a node has state wood or glued wood. In Figure 3 an example for this approach is shown schematically. The wood particle occupies three nodes of the lattice, which is indicated by the brown hexagons. The underlying lattice is depicted by the black lines. The brown circles represent the cells with state wood. These cells are chosen corresponding to the direction of movement, which is a certain lattice velocity, of the wood particle. The states of the other cells have to be empty due to the occupation by the wood particle.

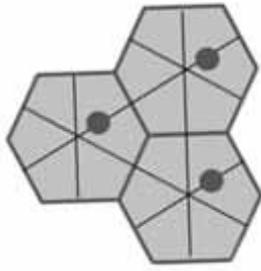


Figure 3: Exemplary illustration of a wood particle, which occupies three nodes.

2.3 Collision

As described in section 2.2 several kinds of states are used. Therefore, the rules for all possible types of collisions have to be defined. In the following, these are illustrated using collisions, where two particles or a particle and a machining part are involved. Of course, it is possible that collision of more than two particles can occur.

First, the collision rule for resin particles is defined. For this collision a perfectly inelastic collision is considered, thus after collision there is one bigger resin particle. In Figure 4 the collision of two resin particles is depicted schematically.

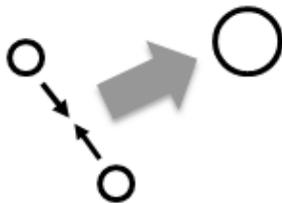


Figure 4: Schematic illustration of the collision of two resin particles.

Next, the collision rule for wood particles is defined. An elastic collision is used for this case, which is shown in Figure 5.

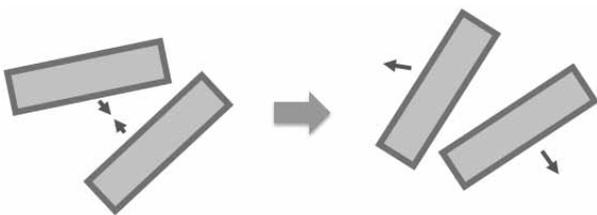


Figure 5: Schematic illustration of the collision of two wood particles.

Due to the porous structure of wood, it can absorb liquids. This is an important fact, which has to be included in the collision rule for wood and resin particles. If such a collision takes place, the resin particle sticks onto the wood particle. Furthermore, a part of the resin particle is absorbed by the wood particle, which is called penetration. In Figure 6 the collision of a wood and resin particle and the penetration are depicted schematically.

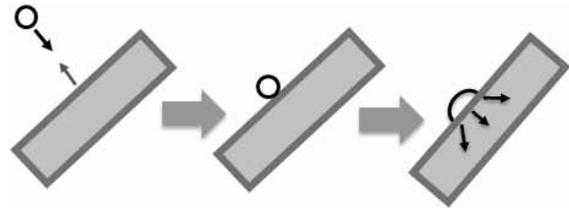


Figure 6: Schematic illustration of the collision of a wood particle and a resin particle.

Next, the collision of a glued wood and an unglued wood particle is considered. This type of collision is modelled as an elastic collision. Furthermore, in this case a part of the resin particle on the surface of the glued wood particle passes to the other wood particle, which is called lubrication effect. In Figure 7 this behaviour is shown schematically. For the collision of two glued wood particles a similar behaviour occurs, i.e. part of the resin drops can pass from one wood particle to the other one.

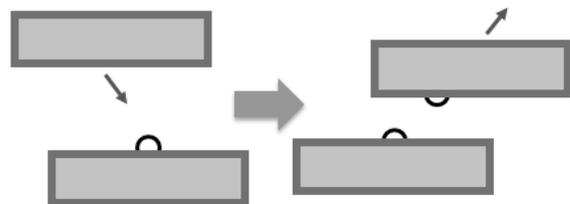


Figure 7: Schematic illustration of the collision of a glued wood and an unglued wood particle.

For all types of collisions mentioned above the velocity of the particles after collision is chosen randomly based on the lattice velocities. This assumption is used for simplification of the implementation.

In the following the collision of a wood particle with parts of the resinating mixer are considered. If a wood particle collides with the wall of the resonating mixer or a mixing arm, an elastic collision takes place. In Figure 8 this type of collision is shown by means of the wall.

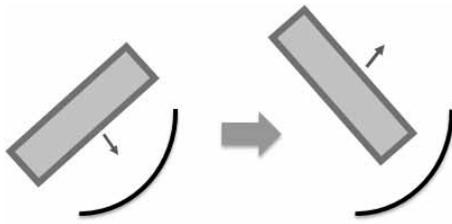


Figure 8: Schematic illustration of the collision of a wood particle and the wall of the resinating mixer.

In contrast to the collision described above, an inelastic collision is used as model for a collision of a resin particle with parts of the mixer. In case of a collision with the wall, the resin particle sticks onto the wall. Therefore, the velocity of the resin particle after collision is zero. In Figure 9 this type of collision is shown. If a resin drop collides with a mixing arm, the velocity of the resin droplet after collision corresponds to the velocity of the mixing arm.



Figure 9: Schematic illustration of the collision of a resin particle with the wall of the resinating mixer.

2.4 Streaming

For the streaming of the wood and resin particles, different approaches are used.

Wood particles. If the ploughshares do not move or do not touch a certain wood particle, the wood particle is stationary. Therefore, in the streaming step the wood particle does not move.

If the velocity is not equal to zero, the cells with state wood move according to the corresponding lattice velocity. Due to the technique described in section 2.2 it is guaranteed, that each node of a specific wood particle moves across the same lattice velocity.

Resin particles. The streaming of the resin particles is modelled by a random walk. Due to gravitational acceleration the resin droplets cannot move upwards. Therefore, not all lattice velocities are allowed for the movement of the particles. A random variable is used for choosing one of the possible lattice velocities. After determination of a permitted lattice velocity, the resin particle moves according to this choice across the lattice.

For a resin particle, which sticks onto the wall, the velocity is zero. If it adheres to a mixing arm, the velocity is set to the velocity of the mixing arm.

3 Simulation

For simplification of the implementation for the two dimensional model, some assumptions were made:

- The milling head and mixing arms are not realized.
- All wood particles have the same dimensions.
- All wood particles, which are not stationary, have the same speed.
- All resin droplets have the same size and velocity until they adhere on a wood particle or a part of the resonating mixer.
- The velocity after collision of the wood particles is determined randomly.
- The penetration of resin into the wood particle is not implemented.

As a further remark, it has to be mentioned, that not all parameters needed for the simulation can be measured by the real system. Therefore, not all parameters correspond to the real system, which has to be included in the interpretation of the simulation results.

The model was implemented using MATLAB R2015b. In Figure 10 a snapshot of a simulation run is shown, whereas the wood particles are shown in brown and the resin droplets in blue.

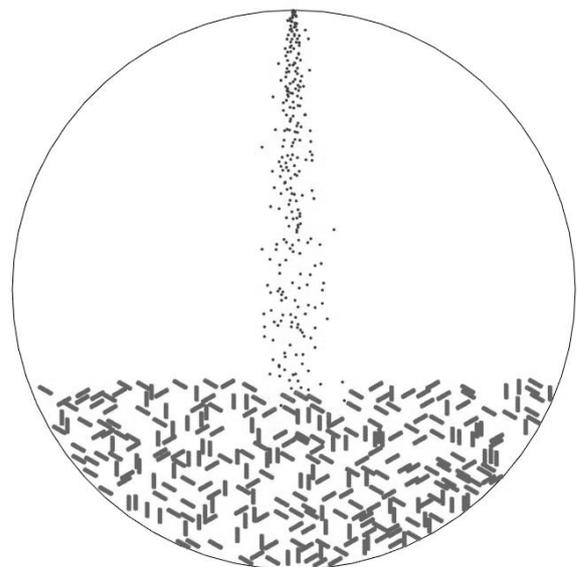


Figure 10: Movement of wood particles (brown) and resin droplets (blue) during simulation.

As illustrated in Figure 10, the resin droplets are moving downwards from the top. Areas of wood particles, which are glued, i.e. a collision of a resin droplet or a glued wood particle and an unglued wood particle took place, are shown in red. For obtaining quantifiable results, the relative frequency of glued wood chips in relation to the total number of wood chips is calculated. First results of this computation are shown in [7].

4 Plans for Validation

It is planned to validate the results using experiments on laboratory scale, since validation is important for the connection between the model and the reality. Regarding the behaviour of wood particles and resin droplets after collision, contact angle measurements will be performed for determining the time dependent behaviour (height, spreading) of the droplet on the wooden surface. In Figure 11 a schematic illustration of the technique for measuring the contact angle θ is shown.

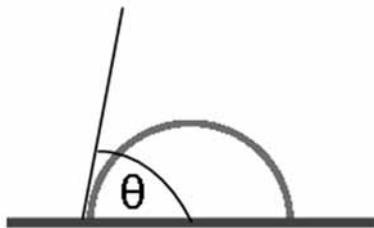


Figure 11: Schematic depiction of contact angle measurement.

Furthermore, for the penetration images of a tiny part of a wood particle, which is glued, can be analysed under a microscope. The lubrication effect will be validated by gluing a wooden board and check the weight before and after it got in contact with an unglued wooden board. The final result of the sub-process gluing can hardly be measured by a defined criterion, therefore the validation of the final simulation results is limited.

5 Conclusion and Outlook

First simulations showed feasibility of the developed mathematical model. The next steps include improving the mathematical model and simulations in two dimensions. Furthermore, the validation of the simulation results will be carried out. Future plans include creating a mathematical model for the process in three dimensions.

References

- [1] Riegler M, Spangl B, Weigl M, Wimmer R, Müller U. "Simulation of a real-time process adaptation in the manufacture of high-density fibreboards using multivariate regression analysis and feedforward control". *Wood Science and Technology*, vol. 47, pp. 1243-1259, 2013.
- [2] Wagenführ A, Scholz F. *Taschenbuch der Holztechnik*. Carl Hanser Verlag GmbH Co KG. 2012.
- [3] Rowell RM. *Handbook of wood chemistry and wood composites*. CRC press. 2005.
- [4] Deppe H, Ernst K. *Taschenbuch der Spanplattentechnik*. DRW, Leinfelden-Echterdingen, 4th edition. 2000.
- [5] Dunky M, Niemz P. *Holzwerkstoffe und Leime: Technologie und Einflussfaktoren*. Springer-Verlag. 2002.
- [6] Wolf-Gladrow DA. *Lattice-gas cellular automata and lattice Boltzmann models: an introduction*. Springer. 2004.
- [7] Rößler C, Riegler M, Breitenecker F. Modeling and simulation of moving wood chips and resin droplets within a resinating mixer using lattice gas cellular automata. *IFAC-PapersOnLine*, Volume 51, Issue 2, 2018, Pages 55-60.

Acknowledgements. The authors gratefully acknowledge the support from the Austrian Research Promotion Agency for funding as well as the Technische Universität Wien and the University of Natural Resources and Life Sciences for providing infrastructure.

Co-Simulation - An Empirical Survey: Applications, Recent Developments and Future Challenges

Gerald Schweiger^{1*}, Georg Engel¹, Josef-Peter Schöggel², Irene Hafner³,

Thierry S. Noudui⁴, Cláudio Gomes⁵

¹ Institute of Software Technology, Graz University of Technology, Inffeldgasse 16b/II, 8010 Graz, Austria; *gerald.schweiger@tugraz.at

² Christian Doppler Laboratory for Sustainable Product Management, University of Graz, Merangasse 18, 8010 Graz, Austria

³ dwh GmbH Simulation Services and Technical Solutions, Neustiftgasse 57-59, 1070 Vienna, Austria

⁴ The United African University of Tanzania, P.O. BOX 36246, Dar Es Salaam, Tanzania

⁵ Aarhus University, Åbogade 34, 8200 Aarhus N, Denmark

SNE 30(2), 2020, 73-76, DOI: 10.11128/sne.30.sn.10516
 Received: December 5, 2019 (Selected MATHMOD 2018 Post-conf. Accepted: Publ.; Accepted: January 20, 2020
 SNE - Simulation Notes Europe, ARGESIM Publisher Vienna, ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. This article describes the first results of an empirical survey on co-simulation conducted with over 50 experts in this field. In the last decades, co-simulation has become an important tool to meet challenges emerging from the increasing complexity of systems and the need for efficient collaboration between experts in different disciplines. However, research on this topic has been motivated by varying fields of interest and developed with different perspectives on application and thus lead to different definitions and emphases within this topic. The present survey aims to clarify some of these different perceptions and open research fields.

Introduction

In recent decades, simulation-driven development has increasingly become established as a central method in industry and academia. This is leveraged by computational advances, like the recent emergence of equation-based modelling languages, which offers new possibilities compared to block diagram modelling using imperative programming languages [16]. Classically, systems are modelled in a single tool, which is referred to as monolithic approach. With the increased complexity of systems and the need for linking several domains in one

model, monolithic approaches have restrictions: sometimes it is not possible to simulate a complex system in a single tool, but even if it is possible, very often there are more suitable tools available for different subsystems. Ideally, every subsystem is modelled in a tool that meets the particular requirements for the domain and the structure of the model. Thus, the need for coupling different tools is a pragmatic one. Co-simulation is an approach to enable a simulation of complex single or multi-domain systems that consists of at least two subsystems (modelled in different tools) which solve coupled (algebraic) differential systems of equations ([5]). An overview of co-simulation approaches and tools, research challenges, and research opportunities are presented e.g. in the references [12, 1, 8, 5, 13]. The proposed empirical survey aims to merge different views of heterogeneous communities which are working in the field of co-simulation, on the state of the art, research gaps and future challenges.

1 Method

As a methodological foundation of the empirical survey, the Delphi method is adopted. The Delphi method is a forecasting technique that bases on the collection and compilation of expert knowledge from a panel of experts in a multi-stage process [3, 6]. It fosters group communication which is intended to deal with complex problems, particularly for the case where there is insufficient knowledge, lack of historical data, or lack of agreement found within the studied field [9]. The Del-

phi method is also conceived to be useful particularly for solving interdisciplinary research problems in a heterogeneous environment [11]. Moreover, it enables determining probable future scenarios. We aim at integrating at least 30 experts in our Delphi study, because despite the lack of a mandatory minimum requirement, [2], for instance, states that 15-30 participants are adequate for studies involving experts with a homogeneous expertise background. For selecting the sample of participants, a Knowledge Resource Nomination Worksheet (KRNW) is used as a guideline [4, 9]. The Delphi study forms two rounds. The first round comprises a mix of open-ended and closed-ended questions. The second round includes only closed-ended questions that are formulated based on the results of the first round. In addition to these standard questions, an additional quantitative analysis of the strengths, weaknesses, opportunities and threats (SWOT) of co-simulation utilizing the Analytic Hierarchy Process (AHP) is conducted. The SWOT-AHP method was introduced by [7] to increase the effectiveness of a primary SWOT analysis as a decision-making tool [10]. In this study, the SWOT-AHP method is utilized to enrich the results of the Delphi study by providing an additional and new perspective on the current state of co-simulation. The questionnaire for the first round of the Delphi study consisted of four parts:

1. *The roots of co-simulation.* This includes questions about different origins for co-simulation, concepts, wording and scientific and industrial communities.
2. *Theoretical questions.* Included are questions regarding the state-of-the-art, research gaps and open issues within continuous, discrete and hybrid co-simulation.
3. *Functional Mock-Up Interface (FMI).* Since FMI is already widely used and it is a promising candidate to become the standard for industry and academia, a section with specific FMI related questions was designed.
4. *Questions related to an overall SWOT analysis of co-simulation.*

At this stage of the survey, the first round of interviews and the expert selection for the second round have both been completed; more than 40 experts have already committed to participate in the second round.

2 Preliminary results

In the first round of interviews, experts had to select three factors for the categories 'Strengths', 'Weaknesses', 'Opportunities' and 'Threats'. In the following, we present the results for the pre-selection of SWOT factors in hierarchical order.

Strengths:

1. Every sub-system can be implemented in a tool that meets the particular requirements for the domain, the structure of the model and the simulation algorithm.
2. Cross-company cooperation is supported (e.g., suppliers and system integrators can exchange virtual "trial components" before signing contracts).
3. Every sub-system can be implemented in a tool that meets the particular requirements for the domain, the structure of the model and the simulation algorithm.

Weaknesses:

1. Computational performance of co-simulation compared to monolithic simulation.
2. Robustness of co-simulation compared to monolithic simulation.
3. Licenses for all programs are required to couple different simulation programs.

Opportunities:

1. Growing co-simulation community / growing industrial adoption.
2. Better communication between theoretical/numerical part, implementation and application/industry.
3. User-friendly tools (pre-defined master algorithms, integrated error estimation, sophisticated analysis to determine best parameterization of solvers and master algorithm).

Threats:

1. Insufficient knowledge/information of users in co-simulation may lead to improper use (e.g. wrong or missing error estimation, stability issues etc.).
2. Lack of exchange/cooperation between theoretical/numerical part, implementation and application/industry.

3. Incompatibility of different standards and co-simulation approaches.

In addition, the experts were asked to name established standards or promising candidates for standards in co-simulation for continuous time, discrete event and hybrid co-simulation. The results show that the Functional Mockup Interface is considered the most promising standard in all three categories (see Figure 1).

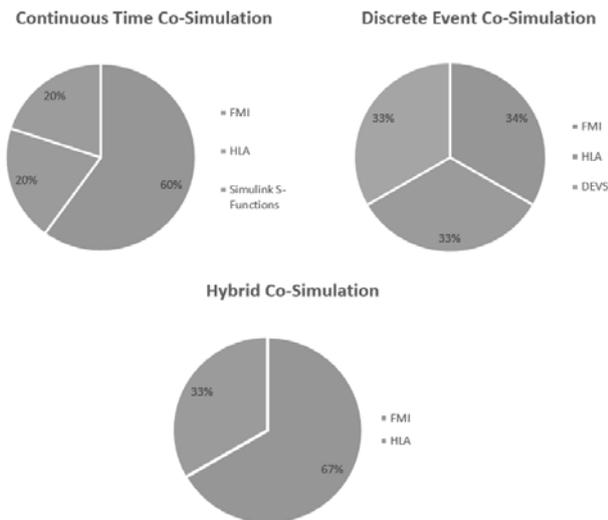


Figure 1: Established standards for co-simulation.

The experts' answers regarding their personal experience of difficulties with co-simulation are summarized in Figure 2.

Co-simulation is an omnipresent topic with ever-varying challenges and openings for new research. An assessment of currently open research fields by the participating experts is shown in Figure 3.

3 Conclusion

The first results of this study confirm the assumption that depending on the primal discipline and field of application, research on co-simulation exhibits different perceptions of the definition, specific focus and importance of challenges as well as open research questions in co-simulation. However, commonalities emerge for certain problems and promising standards, which are to some extent already worked on by experts from different disciplines in international cooperation. Final results of the survey can be found in [14] and [15].

References

- [1] Atam E. Current software barriers to advanced modelbased control design for energy efficient buildings. *Renewable and Sustainable Energy Reviews*. 2017; 73: 1031–1040. doi: 10.1016/j.rser.2017.02.015.
- [2] Clayton, M. J. Delphi: A technique to harness expert opinion for critical decision-making tasks in education. *Educational Psychology*. 1997; 17(4): 373–386. doi: 10.1080/0144341970170401.
- [3] Dalkey N, Helmer O. An Experimental Application of the DELPHI Method to the Use of Experts. *Management Science*. 1963; 9(3): 351–515. doi: 10.1287/mnsc.9.3.458.
- [4] Delbecq AL, Van de Ven AH, Gustafson DH. *Group Techniques for Program Planning: A Guide to Nominal Groups and Delphi Process*. Middleton: Green Briar Press; 1986.
- [5] Gomes C, Thule C, Broman D, Larsen PG, Vangheluwe H. Co-Simulation: A Survey. *ACM Computing Surveys*. 2018; 51(3): 1-33. doi:10.1145/3179993
- [6] Hsu C., Sandford B. The Delphi Technique: Making Sense of Consensus. *Practical Assessment, Research & Evaluation*. 2007; 12.
- [7] Kurttila M, Pesonen M, Kangas J, Kajanus M. Utilizing the analytic hierarchy process (AHP) in SWOT analysis - a hybrid method and its application to a forest-certification case. *Forest Policy and Economics*. 2000; 1(1): 41–52. doi: 10.1016/S1389-9341(99)00004-0.
- [8] Oppelt M, Wolf G, Urbas L. Life Cycle Simulation for a Process Plant based on a Two-Dimensional Co-Simulation Approach. *Computer Aided Chemical Engineering*. 2015; 37: 935–940. doi: 10.1016/B978-0-444-63577-8.50001-2.
- [9] Okoli C, Pawlowski SD. The Delphi method as a research tool : an example , design considerations and applications. *Information & Management*. 2004; 72(1): 15–29. doi: 10.1016/j.im.2003.11.002.
- [10] Reinsberger K, Brudermann T, Posch A. The role of photovoltaics in energy transition - Assessing the prospects for a regime shift. *GAIA - Ecological Perspectives for Science and Society*. 2015; 24(1): 41–47. doi: 10.14512/gaia.24.1.9.
- [11] Stern T, Heil G, Ledl C, Schwarzbauer P. Identifying innovation barriers using a Delphi method approach: the case of technical lignin in the wood-based panel industry. *International Wood Products Journal*. 2012; 3(2): 116–123. doi: 10.1179/2042645312Y.0000000015.

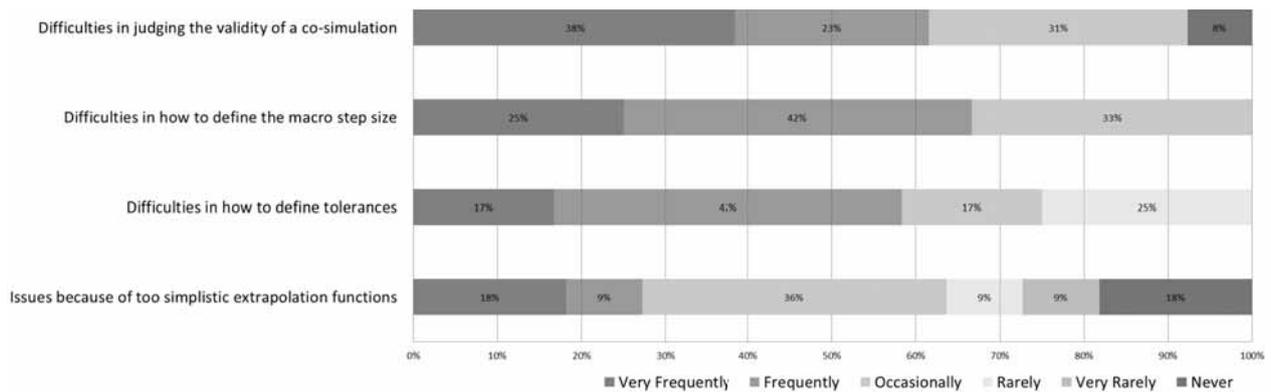


Figure 2: Experts' personal experience with co-simulation.

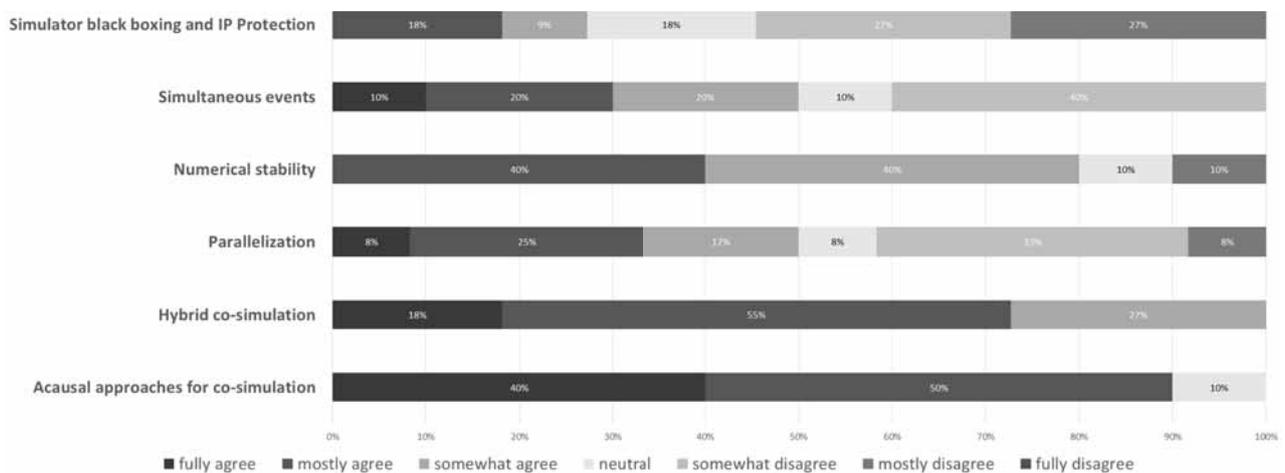


Figure 3: Open research fields in the area of co-simulation.

- [12] Trcka M. *Co-simulation for Performance Prediction of Innovative Integrated Mechanical Energy Systems* [dissertation]. [Faculty of Architecture, Building and Planning, (The Netherlands)]. Technische Universiteit Eindhoven; 2008.
- [13] Hafner I, Popper N. On the Terminology and Structuring of Co-simulation Methods *Proceedings of the 8th International Workshop on Equation-Based Object-Oriented Modeling Languages and Tools*. 2017; Weßling, Germany. doi: 10.1145/3158191.3158203.
- [14] Schweiger G, Gomes C, Engel G, Hafner I, Schoeggel J, Posch A, Nouidui T. Functional Mock-up Interface: An empirical survey identifies research challenges and current barriers. *The American Modelica Conference 2018*. 2018 Oct; Somberg Conference Center, Cambridge MA, USA. doi: 10.3384/ecp18154138.
- [15] Schweiger G, Gomes C, Engel G, Hafner I, Schoeggel J, Posch A, Nouidui T. An Empirical Survey on Co-Simulation: Promising Standards, Challenges and Research Needs. *Simulation Modelling Practice and Theory*. 2019; 95(issue): 148–163. doi: 10.1016/j.simpat.2019.05.001.
- [16] Schweiger G, Nilsson H, Schoeggel J, Birk W, Posch A. Modeling and simulation of large-scale systems: A systematic comparison of modeling paradigms *Applied Mathematics and Computation*. 2020; 365. doi: 10.1016/j.amc.2019.124713.

MATLAB-based Modelling and Simulation of Forest Development: Growth – Harvesting – Draught – Bark Beetle – Biomass Plant Usage

Constantin Königswieser*, Anicia Jahn

Inst. of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8-10,
1040 Vienna Austria; Constantin.Königswieser@student.tuwien.ac.at

SNE 30(2), 2020, 77 - 84, DOI: 10.11128/sne.30.stn.10517
Received: April 2, 2020; Revised May 5, 2020;
Accepted: May 10, 2020
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. The present study addresses modelling and effective MATLAB simulation of forest of growth of a forest. The modeling of the forest growth was conducted using certain assumptions on growth. Besides the simulation the aim of the study was to simulate as many individual trees as possible with the limited processing power of a common notebook. The developed simulation algorithm calculates many individual objects – trees – with the main goal to receive a fast and efficient program. This was achieved via the ideas from cellular automata.

Introduction

Global warming has led to many environmental problems, such as increased periods of drought and less precipitation. The climate change will affect various types of animals and plants and change the growth dynamics of forest trees. This will change Austrian landscapes permanently. Especially spruce, which is very common in Austria's forests, will cease in higher temperatures.

According to this, certain steps have to be made to ensure the health of our forests [1]:

- developing tools for the assessment of ecosystem sustainability
- studying the quantitative and qualitative patterns of ecosystem dynamics under the impact of global warming
- establishing new paradigms of sustainable forest management

This prospect requires effective predictions of forest growth and the dynamics of its basic characteristics as well as the implementation of disturbances, which can be considered as the key drivers of forest ecosystem dynamics because of their significant influence on its structure and functioning. To ascertain an effective prediction, a simulation model, namely an imitation of a real life system, can be developed [1].

This study addresses the application of a forest simulation using MATLAB. Besides the actual simulation of a spruce forest, the aim of this study was to simulate as many individual trees as possible with the limited processing power of a common notebook. The boundaries of MATLAB were tested by developing an algorithm calculating many individual objects with the main goal to develop a fast and efficient program. This was achieved via ideas from cellular automata, which will be discussed in Chapter 2.1. We used MATLAB versions R2016a and R2019b to build and visualize our model.

As mentioned before we model a naturally growing forest with as many trees as possible to test the boundaries of MATLAB. The goal is to find out if such a program would be feasible on a common notebook. Different forest sizes are tested on a notebook with the following properties: Intel Core i7-4710HQ, 2,5 GHz, 16GB RAM.

Furthermore, to assess the influence of different scenarios, four general simulation scenarios were designed, which are the following:

- Contamination
- Harvest
- Drought
- Bark beetles
- Biomass plant usage

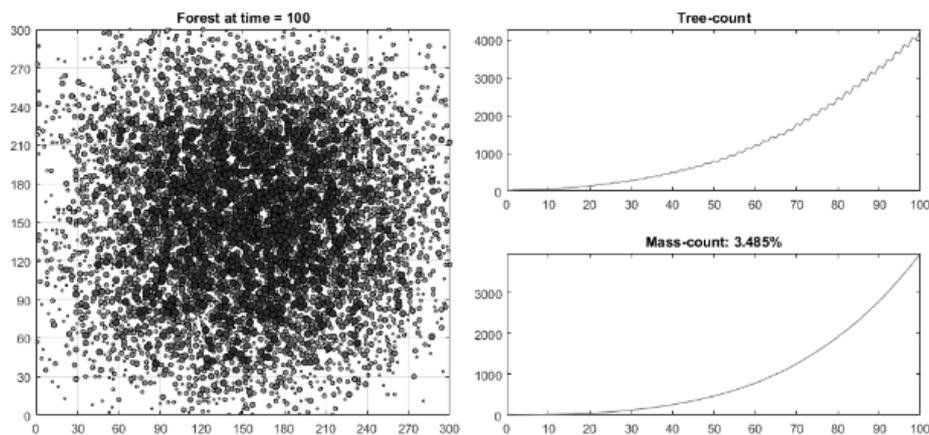


Figure 1: Standard simulation diagrams, which are updated each time step. Left: 2D-presentation of the forest. Right: number of trees and wood mass of the forest.

1 Modelling

1.1 Forest modelling

The first thing that comes to mind, when modelling a forest with individual trees, would be object orientated programming, which MATLAB offers but is not suited for in large scales as the program is an analysis software. So instead of creating an object structure we use MATLAB's matrix-based data structure to store the different properties of every single tree in corresponding cells.

This model was inspired by the cellular automata, a discrete, abstract computational system [2]. In these automata an occupied space is represented by one, otherwise it is zero. Similar to cellular automata our model is discrete; however, as a cellular automaton is deterministic and we introduce probabilities, our model cannot be classified as such.

We generate a 2D discrete space (see Figure 1) where each cell's future state is dependent on its current state and neighbors, namely depending on position.

One of the major questions arising, when developing a model, is the level of detail and processing resolution. Building any simulation model, a number of assumptions and simplifications have to be made.

We assume a flat and confined area with unchanging weather conditions. For this model, two main assumptions are made. The first is that the quantity of neighbors and the tree's age is relevant for spreading. The second assumption is that trees can randomly die depending on age. In this random number, influences such as light, water, nutrition and air are included.

For this project, a variable square forest area is populated with 20 trees, age 10, as initial condition. The planting locations are randomly chosen within a radius of 20 m around the central point of the area. Different simulation scenarios are conducted, taking into account both the regime of forest management, namely harvest, as well as natural disturbances, such as droughts. Certain parameters, such as spreading, death rate etc. of a spruce tree, are necessary for model initialization.

Structure of the forest. For this kind of object-orientated programming it would be obvious to populate a forest object with tree objects in which all properties, such as age, mass, number of neighbors etc., are stored. As MATLAB is not suited for this kind of programming, we chose a different approach.

We generate only one object (class as it is called in MATLAB), namely the forest, in which all properties are stored in separate matrices. Inspired by the aforementioned cellular automata, the size of the forest determines the size of the matrix so that each entry corresponds to a location in the area. Furthermore we assume that one entry matches one square meter. For example, a forest of size 100 m \times 100 m generates a 100 \times 100 matrix.

Since the matrices provide us with a grid, there is no need to compare each tree with every other tree. Instead we only need to check the cells within a given radius for neighboring tree, which reduces the number of operations needed dramatically.

Methods	Descriptions
<i>Forest()</i>	Construction-method of the forest. Initializes all matrices for storing properties.
<i>findTrees()</i>	Searches for all trees in the given area and stores their x- and y-location in a 2xn-matrix, where n is the number of found trees (namely <i>foundTrees</i>).
<i>findNeighbours()</i>	Loops through all found trees and counts all neighbors in a given radius.
<i>plantTree()</i>	Plants a tree, with a given age, by changing the corresponding entry of the tree-matrix to 1.
<i>markTrees()</i>	Loops through all found trees and applies the rules for dying for each tree and marks those which will be removed in the next time step.
<i>removeTree()</i>	Removes all marked trees and resets the properties of the given location.
<i>spread()</i>	Loops through all found trees, generates a random location for the sapling and checks afterwards, depending on neighbor count, if this sapling can be planted.
<i>aging()</i>	Increases the age of the tree by one as well as the mass. Additionally it increases a color counter every 5 years by one; up to a maximum of 10.
<i>die()</i>	Finds all marked trees and calls the <i>removeTree()</i> -function.
<i>harvest()</i>	Loops over the current harvest area and marks all trees with in that area over a given age and records the harvested mass.
<i>infest()</i>	Finds all trees within a given (small) area and choses randomly a tree and increases the beetle count by 20 on the chosen trees (100 times).
<i>beetleSpread()</i>	Finds every infested tree and loops over them. It multiplies the number of beetles by a given factor and spreads the number of beetles evenly on neighboring trees.
<i>boundaries()</i>	A helpful function which generates the boundaries of the search area with a given tree-location and radius, including edge cases.
<i>draw()</i>	Generates a scatter-plot of all trees with full circles of different sizes and shades of green depending on age (see Figure 1).
<i>drawMarked()</i>	Generates a scatter-plot of all marked trees with red crosses.

Table 1: Summary of the forest's methods.

To update the forest each year, a number of different functions (methods as they are called in MATLAB classes) are used, given in Table 1.

To retrieve the indices, the find function is heavily used. The find function returns the indices and values of nonzero elements and thus allows us to loop over occupied spaces only.

Similar to cellular automata we appoint a few simple rules:

- Spreading: depending on the number of neighbors the tree plants a sapling in a randomly generated unoccupied space in a certain radius; however, not too close to other trees
- Aging: every year the tree grows older and accumulates mass depending on age
- Dying: each tree has a certain probability to die, depending on its age

Spreading: Every other year, as it is the case in nature [3], the trees spread and plant new saplings. This is achieved by appointing different probabilities according to different numbers of neighbors in a certain distance; for this study a radius of 10 m is chosen.

- zero neighbors: no spreading: 0%
- less than or equal to three neighbors: 50% chance to spread
- between three and ten neighbors: 100% chance to spread
- greater than or equal to ten neighbors: 50% chance to spread
- more than fifty neighbors: no spreading: 0%

These probabilities are conducted according to the following considerations: if the tree has no neighbors, there is no chance to pollinate the tree. If it has a certain amount of neighbors there are enough pollen to fertilize the seeds.

However, if the tree has too many neighbors there is not enough space for the seedlings to grow and no spreading occurs.

Aging: every year, the tree grows older and accumulates more mass depending on age. Till the age of 70, we assumed a linear growth rate of 28 kg/year. From 70 to 140 the tree has a growth rate of 14 kg/year until it reaches its final mass of approximately 3.000 kg. When calculating the mass, we assume that the harvested wood has already sufficiently dried with a water content of 20%. [4]

Dying: Each tree has a certain probability to die, according to its age.

- Less than or equal to three years: 5% chance to die
- Greater than three years: 1% chance to die
- Greater than 150 years: 20% chance to die

With the probability to die each year, general stress-factors such as diseases, damage by wildlife, or storm losses are considered. The effects of drought and bark beetles are dealt with separately. Younger trees and very old trees are naturally more susceptible to these general damages.

As a forest is a very complex ecosystem, data on these specific parameters and probabilities is missing. As a result, the authors decided to conduct the procedure of model initialization by using the following parameters:

- trees grow on an area of one hectare [5]
- The forest has a growth rate of 2,5 % cubic meter each year [6]

To arrive at these targets, we adjusted the spreading and neighbor radii by means of computational experimentation.

1.2 Contamination modelling

The first issue is the implementation of a contaminated area in the forest, where trees can only survive for a short amount of time as there are toxic chemicals in the ground, which prevent the trees to grow further.

This is achieved by implementing a given area, which is marked as “contaminated” where trees, which grow on this ground, will cease after two years. This implementation could be used for rivers or roads as well.

We preset a matrix, namely contaminated, with the elements one and zero for contaminated and non-contaminated areas respectively. In the function markTrees() every tree is checked if the location is considered contaminated and if so, it will die if it is older than three years.

1.3 Harvest modelling

The second issue is harvesting specific areas, which is conducted by dividing the whole area in ten equal strips. At a given time, all trees over a given age in one strip (0,9 ha) are harvested. In the function harvest(), each entry of the strip is checked for trees. Then, each tree reaching the requested age is marked and subsequently removed. Furthermore, the mass count is recorded. Each iteration (namely one year), the harvesting strip shifts to the next starting point, which is conducted each year till it reaches the beginning and starts once again.

1.4 Drought modelling

Less precipitation causes increasing stress for the trees which results in a higher mortality rate and reduced mass growth as well as fewer offspring [7]. We assume an increased mortality rate of 20% and a growth reduction of 50% and a reduced chance of generating offspring by 50%.

To implement this natural phenomenon, these rates are changed in the functions markTrees(), spread() and aging() depending on a drought flag, which is set for a given time period.

1.5 Bark beetle modelling

The last issue is the bark beetle infestation which will be a great threat in the future, since global warming causes warmer winters and therefore increases their chances to survive.

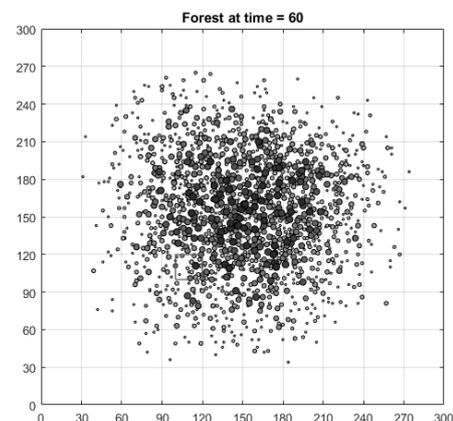


Figure 2: Red box: Initial infestation area.

The biggest problem of these beetles is the explosive growth rate and the fact that the trees succumb to these immense numbers [8].

In our model the beetles spread three times a year and infest each tree without exception (such as age etc.). The number of beetles on each infested tree multiplies tenfold and the same amount is equally distributed to neighboring trees within the radius of 10 m. Furthermore, each tree with 200 beetles dies and the beetles with it.

To initiate the infestation the function infest() injects a given number of beetles into a certain area of the forest once (see Figure 2). To spread the infestation the function beetleSpread() is called three times per time step. The markTrees()-function checks the number of beetles on every tree and marks them, if the given threshold is met.

2 Simulation Results

At first to verify the model and to meet the set targets a 9 ha forest (300x300) is simulated five times and the mean values are calculated. The results are shown in figures 3 and 4. The figures show that the target tree population of 1000 trees per ha is met in its equilibrium state. This state is reached after 250 years.

The forest of this model follows a logistical growth and thus has a changing growth rate. For the second target, we estimate the growth rate by taking the mean value between the years 50 and 200, which results in a mean growth rate of 2,67% m³ per year, which corresponds well with the set target of 2,5% m³ per year.

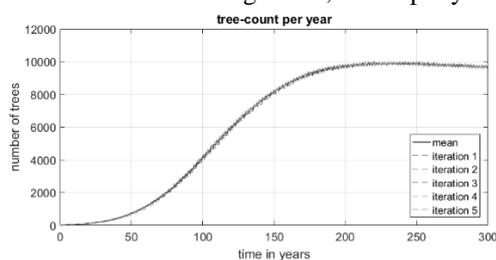


Figure 3: 5 iterations and mean value (blue) of the forest's tree count each year over time span of 300 years.

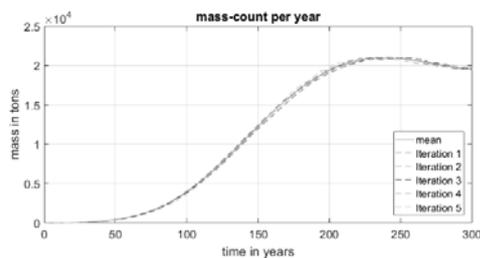


Figure 4: 5 iterations and mean value (blue) of the forest's wood mass each year over time span of 300 years.

As pictured the deviation from the mean value is small enough to justify the low number of iterations. Thus every following scenario (contamination, harvest, drought and bark beetle) is conducted in the same way, by iterating five times and taking the mean. Additionally other scenarios are deactivated, to examine each scenario by itself.

2.1 Contamination

As Figure 5 shows, the contaminated area in the forest remains blank, except for a few trees of age 1-3 years.

The contaminated area acts as a barrier and the forest has to grow around it. Therefore the forest spreads more slowly behind the contaminated area.

As expected the growth is delayed and terminates in a smaller total amount of trees and mass compared to the forest without a contaminated area (see Figures 6 and 7).

2.2 Harvesting in strips

In this simulation harvesting starts after 100 years and lasts for 150 years and only trees older than 90 years are felled, since the ideal age for harvesting of spruce is between 80 and 120 years. [9]

The mean value of the harvested wood over this time span is 116,1 tons per year. The first 50 years of harvesting have considerably lower yields, as in the beginning the forest is still growing in size and only few trees are over 90 years old.

However, after 95 years of harvesting the forest is already at its fullest with 9.000 trees and therefore a mean harvest of 188,32 tons per year is guaranteed.

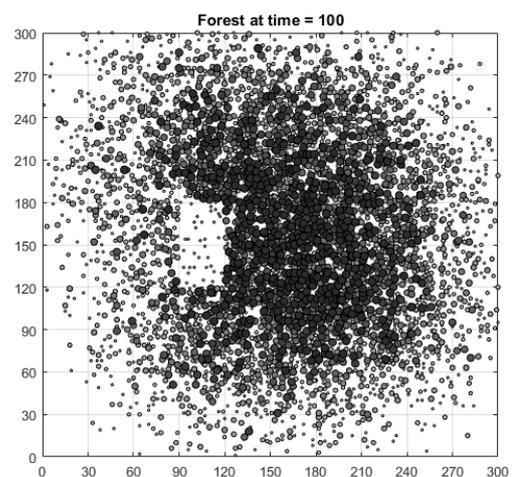


Figure 5: Forest at the age of 100 years with a contaminated area between (90,120) and (120,180).

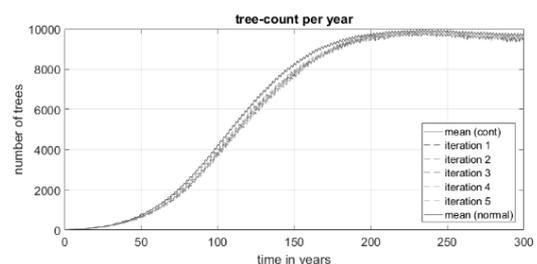


Figure 6: Tree count of the whole forest with contaminated area.

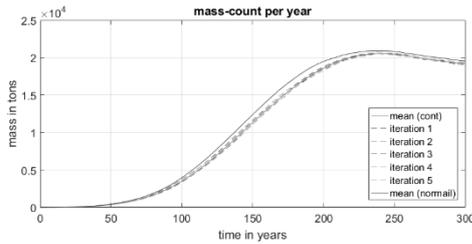


Figure 7: Mass of whole forest with contaminated area.

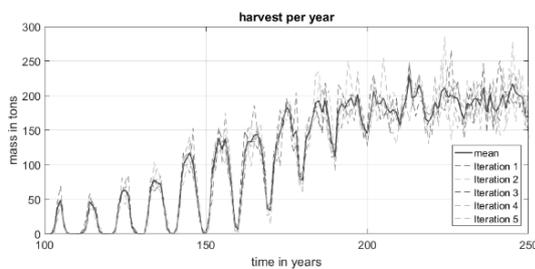


Figure 8: Five iterations and the mean value (blue) of the harvested wood over a time span of 150 years.

As shown in Figure 9 and Figure 10 the forest can maintain a steady growth rate and can sustain itself without problems.

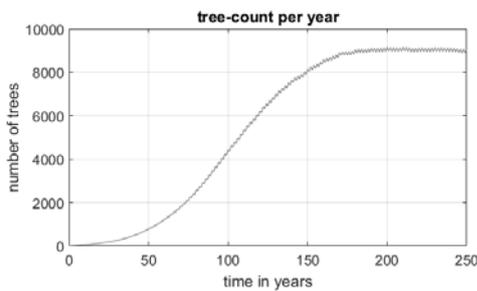


Figure 9: Tree count of the whole forest while harvesting (mean value over 5 iterations).

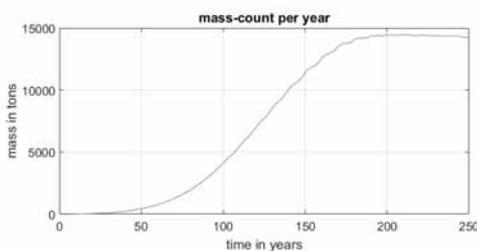


Figure 10: Mass count of the whole forest while harvesting (mean value over 5 iterations).

2.3 Household energy supply by wood

‘How many households can be supplied with electric energy using the felled wood?’

This question arises, how much energy can be gained using 116,1 tons of felled wood. An average household in Austria consumes 4.415 kWh electric energy each year [10]. Assuming, that the wood is already sufficiently dry with a water content of 20%, the wood has a fuel value of 4 kWh per kg [11]. Consequently, transferring these 116,1 tons of dry wood to a biomass power plant, it would amount to 464.400 kWh which could provide approximately 105 households with electric energy in Austria each year.

2.4 Drought

The drought is implemented twice, after 50 and 250 years, each period lasts for ten years.

The model shows that a drought at an early stage has no significant impact, the growth rate merely stagnates. The drought at the latter period, with a full grown forest of 9.000 trees, causes an apparent reduction of growth rate, as well as a higher mortality. The forest seems to recover after a certain amount of time, however it takes more than 100 years to regain the same amount of trees as well as wood mass.

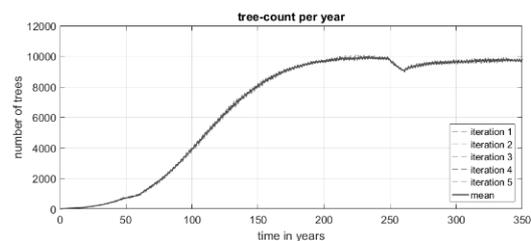


Figure 11: Mass count per year of drought periods at 50 and 250 years over a time span of 350 years.

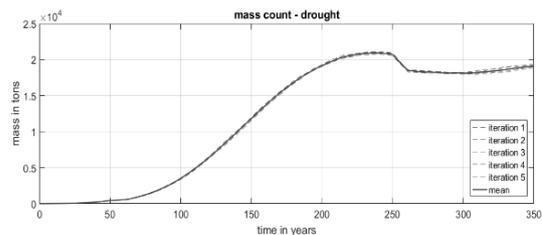


Figure 12: Mass count per year of drought periods at 50 and 250 years over a time span of 350 years.

2.5 Bark beetle

To investigate under what conditions a bark beetle outbreak will occur, we conduct nine scenarios in which we differ the year of infestation. 2.000 bark beetles are inserted in a specified (de-central) area (see Figure 2) starting in the year 20, each time incrementing by 5 up to 60 years.

As Figures Figure 13 and Figure 14 show, the first bark beetle outbreak happens in the year 40. Before that, the bark beetles couldn't spread, as the number of trees in the specified area is too small (see Figure 15). The outbreak lasts approximately five years and eradicates nearly all of the tree population, as a consequence the beetles go extinct since there are too few trees left and the distance between these trees is too high.

Even though the bark beetles cause the forest population to decrease immensely and delay its growth by ten years, it is always able to recover within approximately 150 years. The simulation-data shows an overshoot in mass as well as tree count 200 years after the outbreak, which may be explained by the following reasoning:

- At the time of the outbreak the trees occupy a far spread area
- After the outbreak only saplings remain

These factors cause a very homogenous forest of widespread, same aged trees. Since in this model the mortality rate is increased significantly (by 19%) at the age of 150 an over proportional amount of trees die and therefore mass and tree count drop. This phenomenon can be especially well observed in the mass count.

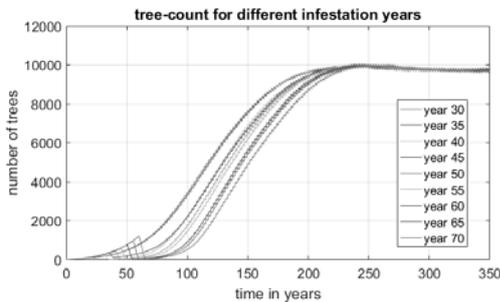


Figure 13: Tree count per year during the bark beetle infestation over a time span of 350 years.

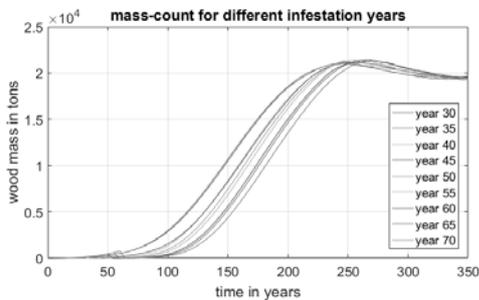


Figure 14: Tree count per year during the bark beetle infestation over a time span of 250 years.

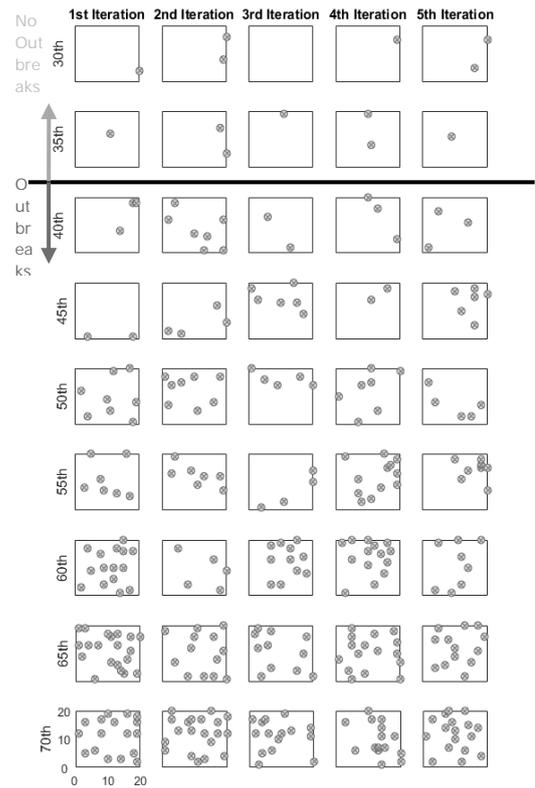


Figure 15: Close-Up of the infestation area (as seen in Figure 1) for every simulation. The rows depict the different years in which the forest is infested.

We assume that each tree killed by bark beetles is removed and as a consequence the beetles from this tree cannot spread further. By doing so, the bark beetles eventually go extinct. Transferring this circumstance to forestry, infested and dead trees have to be removed in order to defeat the bark beetle.

As Figure 15 shows the outbreak only occurs if the trees have enough neighbors, namely more than two. This would suggest that a forest with less planting density is less prone to bark beetles.

2.6 Computational power

To identify the computational power on a notebook, different forest sizes were simulated, starting from 1 ha to 64 ha. The results are shown in Figure 16, and it can be clearly seen that the simulation time grows linearly to the side length of the forest area.

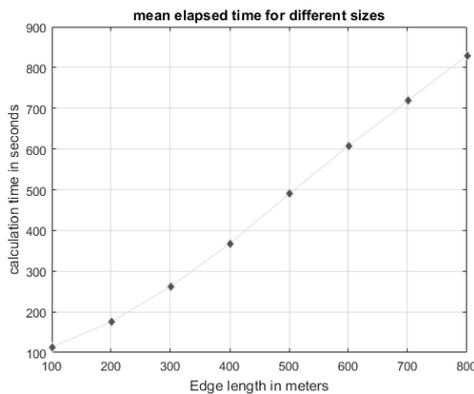


Figure 16: Mean time elapsed for different forest sizes.

3 Conclusion

This study shows, that complex systems with a considerable amount of data and computational power are able to run on a simple notebook using MATLAB, if the program is implemented with certain computational effort saving techniques, such as avoiding nested loops – in this regard the `find()`-function provided by MATLAB was profoundly useful.

According to this model it is possible to harvest wood without constraining the forest's natural growth, if certain rules are followed. Here the harvest took place in strips and only trees older than 90 years are felled. Other harvesting techniques may be examined in future studies. Finally it should be mentioned that a yearly reliable yield can only be achieved, if the forest has reached nearly its maximum size.

Likewise, the forest size respectively the planting density plays a crucial role in the bark beetle outbreak. According to this model, the outbreak can only occur, if there are enough neighboring trees around the infestation center. Though if the density is high enough, the forest will be nearly annihilated within less than 5 years.

However the forest always recovers and returns to its previous size in less than 50 years. It has to be considered that the forest would only survive a bark beetle invasion, if the infested and dead trees are removed and thus eradicate the bark beetles.

The ideas of the cellular automata proved to be very viable to depict a very complex system in a comparably simple way. Over all this model manages to generally resemble a naturally growing forest with two very simple rules.

To receive more realistic results, nonlinear interactions between the individual trees and their abiotic environment, such as light, air and soil, have to be considered, which could be implemented by adding more complex rules.

Acknowledgement

This study is a student's work for preparation of a master thesis (TU Wien, Institute of Analysis and Scientific Computing; supervisor: Prof. Dr. Felix Breitenecker *Felix.Breitenecker@tuwien.ac.at*). This is the first *SNE Student Note*, a new type of SNE notes introduced in 2020.

References

- [1] Shanin VN, Komarov AS, Bykhovets SS. *Simulation modelling for sustainable forest management: a case study*. Procedia Environmental Sciences; 2012, Vol. 13.
- [2] *Stanford Encyclopedia of Philosophy*. [Online]. Available: <https://plato.stanford.edu/entries/cellular-automata/>. [Accessed 8 March 2020].
- [3] *tierfreund*. [Online]. Available: <https://www.tierfreund.de/waldkiefer/>. [Accessed 8 March 2020].
- [4] *Handwerkerratgeber*. [Online]. Available: <https://handwerkerratgeber.info/spezifisches-gewicht-von-holz-bestimmen/#>. [Accessed 8 March 2020].
- [5] *Pro Holz* [Online]. Available: <https://www.proholz.at/zuschnitt/51/der-oesterreichische-wald>. [Accessed 8 March 2020].
- [6] *Holz ist genial*. [Online]. Available: <https://www.holzistgenial.at/blog/10-fakten-zum-wald-in-oesterreich/>. [Accessed 8 March 2020].
- [7] *Waldwissen*. [Online]. Available: https://www.waldwissen.net/waldwirtschaft/schaden/trockenheit/bfw_trockenstress/index_DE. [Accessed 8 March 2020].
- [8] *Waldverband*. [Online]. Available: <https://www.waldverband.at/wp-content/uploads/2016/06/Borkenk%C3%A4fer-2016-web-1.pdf>. [Accessed 12 March 2020].
- [9] *wald-prinz*. [Online]. Available: <http://www.wald-prinz.de/umtriebszeit-wie-lange-benotigt-ein-baum-bis-zur-hiebsreife/3697>. [Accessed 14 March 2020].
- [10] *Österreichs Energie*. [Online]. Available: <https://oesterreichsenergie.at/daten-fakten-zum-stromverbrauch.html>. [Accessed 8 March 2020].
- [11] *Kesselheld*. [Online]. Available: <https://www.kesselheld.de/heizwert-hackschnitzel/>. [Accessed 8 March 2020].

SNE Simulation News

EUROSIM Data and Quick Info




VESS – Virtual EUROSIM Seminar
Virtual Simulation Presentations, from June 2020 on www.eurosim2023.eu




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




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
EUROSIM Conferences and 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 Tanzler (Layout, Organisation)
Irmgard Husinsky (Web, Electronic Publishing)
Felix Breitenecker 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 professionalists (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 Breitenecker felix.breitenecker@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. Breitenecker, felix.breitenecker@tuwien.ac.at
Repr. EUROSIM	F. Breitenecker, felix.breitenecker@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 *IBM* 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@unica.it
Edit. Board SNE	F. Longo, f.longo@unica.it
Web EUROSIM	F. Longo, f.longo@unica.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	Zdzislaw 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.ed.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, Kjølnes 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>mradulescu.csmro@yahoo.com</i>
Repr. EUROSIM	Marius Radulescu, <i>mradulescu.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
- FRANCOSIM – 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 Breitenecker

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 Breitenecker

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:

The screenshot shows the 'Manuscript Submission' form on the SNE website. At the top, the logo 'SNE SIMULATION NOTES EUROPE' is displayed. Below the logo, a navigation bar includes links for 'Home', 'Aims and Scope', 'Editorial Board', 'SNE Volumes', and 'Search'. The main content area is titled 'Submission' and 'Submission Confirmation'. A breadcrumb trail reads 'SNE > Contribute / Contact > Submission'. The 'Manuscript Submission' form includes the following fields:

- Article Title: *
- Type: (Dropdown menu with 'Technical Note' selected)
- 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

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

MATLAB®
The language of technical computing

