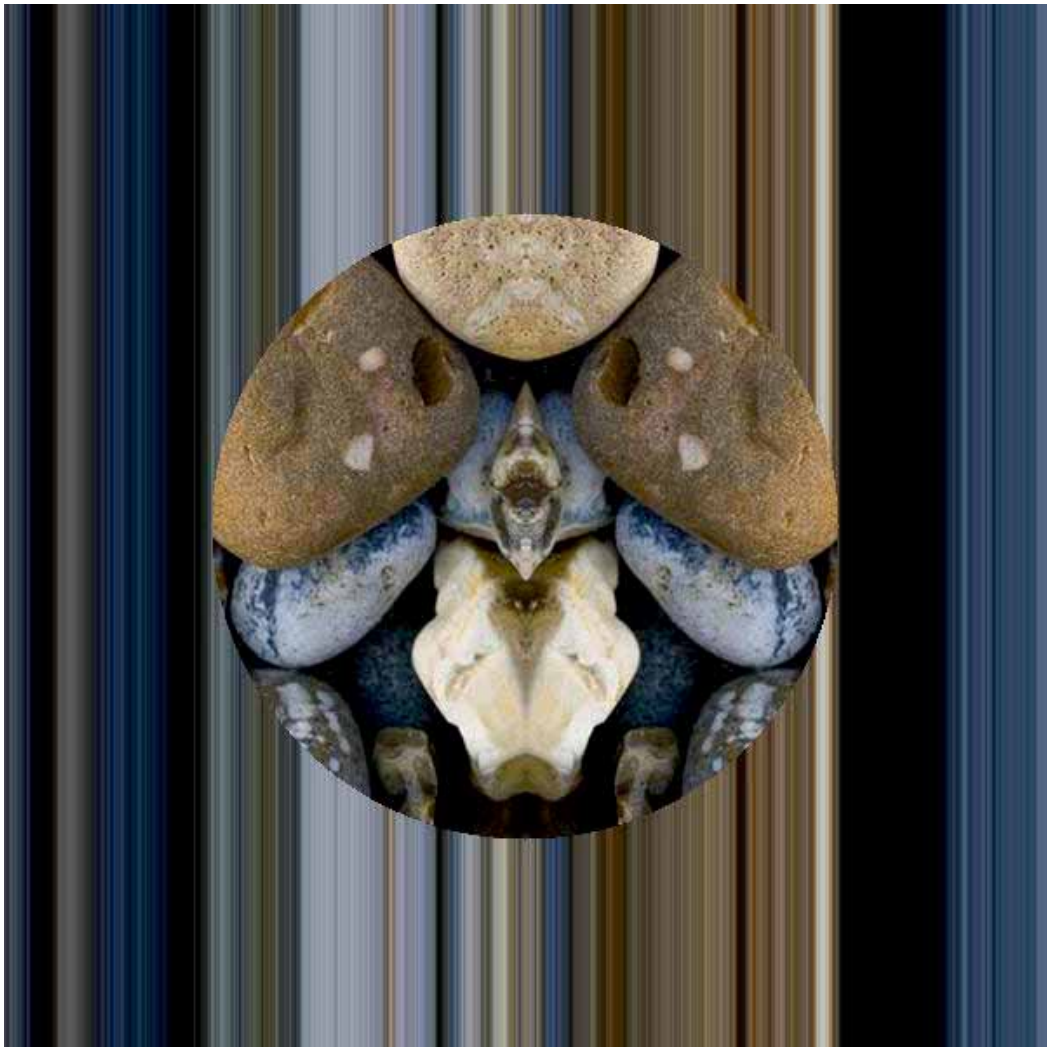
SNE SIMULATION NOTES EUROPE



Journal on Developments and Trends in Modelling and Simulation

EUROSIM Scientific Membership Journal

Vol. 32 No.4, December 2022 ISSN Online 2306-0271 DOI 10.11128/sne.32.4.1062
ISSN Print 2305-9974 ISBN Print 978-3-903311-33-6

TU Verlag an der Technischen Universität Wien

ARGESIM

Das alljährliche Workshop der Fachgruppen Simulation technischer Systeme, Grundlagen und Methoden in Modellbildung und Simulation, Edukation und Simulation wird am 6. und 7. März 2023 in Magdeburg an der Otto-von-Guericke-Universität stattfinden.

Wir hoffen wieder auf regen Austausch und eine offene Atmosphäre für lebendige Diskussionen und den Informations- und Erfahrungsaustausch zwischen Fachleuten aus Hochschulen, Forschungseinrichtungen und der Industrie.

Beiträge können sowohl anwendungsorientiert als auch grundlagenorientiert sein. Aber auch die Vorstellung von Ideenskizzen oder noch laufenden Projekten sind willkommen und müssen nicht unbedingt den vorgegebenen thematischen Schwerpunkten entsprechen.

www.asim-gi.org/magdeburg2023/

The EUROSIM Congress 2023, the **11th EUROSIM Congress**, will be organized by DBSS, the Dutch Benelux Simulation Society, in Amsterdam, July 3-5, 2023.

This time, the EUROSIM Congress's theme is ***Simulation for a sustainable future***. And there is no better place than Amsterdam to celebrate the link between simulation and sustainability. You will be pleased to discover the fascinating and vibrant city of Amsterdam, a hub for many sectors such as logistics, ICT, art, financial and business services, and fashion, among others.

www.eurosim2022.eu

Als größte europäische Simulationstagung für Produktion und Logistik präsentiert die ASIM Fachtagung alle zwei Jahre zukunftsweisende Trends und aktuelle Entwicklungen, wissenschaftliche Arbeiten sowie interessante Anwendungen in der Industrie.

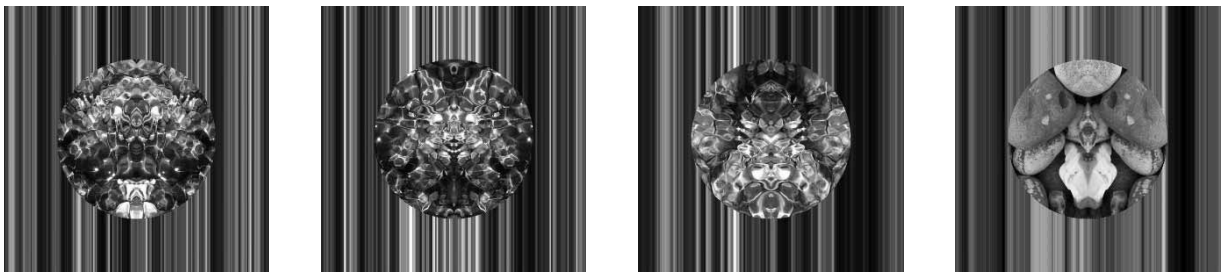
Der thematische Fokus der nächsten Fachtagung lautet Nachhaltigkeit in Produktion und Logistik. Damit greift die Tagung eine wichtige gesellschaftliche Thematik auf und fokussiert gleichzeitig auf aktuelle Forschungsthemen der Simulationswelt.

www.asim-fachtagung-spl.de/

Editorial

Dear Readers, In this issue SNE 32(4) two very interesting Overview Notes on issues of fitness for train simulation models by D. J. Murray-Smith, and on methods for integrated simulation by N. Popper, frame Technical Notes and Benchmark Notes.
While A. Andreasen et al. present a tool for calculating CO2 pipeline wave speed, P. Junglas et al. sketch implementation of examples with NSA-DEVS, and Brunner & Hametner review recommender systems in the medical domain, and P. Bauerstätter introduces into COMPASI-based compartment modeling for diets effects. And we are very glad, that the series of ARGESIM Benchmarks Notes is continued by a parallel and distributed simulation' with Open MPI/GSL and Matlab PCT for Monte-Carlo and PDE case studies by D. Jammer et al. – one of the rare contributions for the 2nd Parallel Benchmark.

For SNE Volume 32, Vlatco Čerić, Past President of the Croatian Simulation Society, provided his artwork as cover pictures for SNE covers, graphics from his series AMULETS. After Amulet 3, 4 an5 this issue completes the series with Amulet 1 (below).



I would like to thank all authors for their contributions in 2022, and thanks to the editorial board members for review and support, and to the organizers of the EUROSIM conferences for co-operation in post-conference contributions. And many thanks to the SNE Editorial Office for layout, typesetting, preparations for printing, electronic publishing, and much more. And again, we are glad, that after the series of virtual conferences we can promote again EUROSIM face-to-face conferences, the EUROSIM Congress 2023, July 2023 Amsterdam, the conference ASIM SPL 2023, Ilmenau, in September 2023, and others (see covers).

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

Contents SNE 32(4)

Online SNE 32(4), DOI 10.11128/sne.32.4.1062
ARGESIM Publisher, Vienna, www.argesim.org
Print SNE 32(4) ISBN 978-3-903311-33-6
TU Verlag Vienna, Print-on-Demand, www.tuverlag.at

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 Breitenecker (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 Breitenecker, TU Wien, Math. Modelling Group ✉ *Felix.Breitenecker@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*

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.
- **BNE** 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:

Issues of Fitness for Purpose in Train Simulation Models: a Review

David J. Murray-Smith*

James Watt School of Engineering, University of Glasgow, Glasgow G12 8QQ, Scotland, United Kingdom;
David.Murray-Smith@glasgow.ac.uk

SNE 32(4), 2022, 177-186, DOI: 10.11128/sne.32.on.10621
Received: 2022-06-19; Revised: 2022-11-04
Accepted: 2022-11-08
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Many simulation models representing the longitudinal dynamics of a train are based on a single point-mass description. This leads to a second-order nonlinear ordinary differential equation, together with algebraic relationships. More complex multi-mass models may be used for models representing long trains involving many separate vehicles. However, in both cases, accuracy is limited by important underlying assumptions, approximations and parametric uncertainties. Another important aspect of train models concerns the direction of information flow. Input variables within conventional train models may represent power or tractive force, with acceleration, speed and distance travelled as output variables. However, inverse simulation methods can also be used, with the required speed or distance as inputs and tractive force, power, or energy as outputs. This allows energy requirements to be established for a given schedule and is useful when investigating fuel or energy economy. Inverse methods can also be used in powertrain design, such as for hybrid hydrogen fuel-cell/battery-electric trains. Issues of fitness for purpose are important in all such applications, both in terms of model uncertainties and in the additional insight offered by inverse simulation methods.

Introduction

In engineering applications, modelling and simulation methods allow early consideration of design trade-offs and system integration issues before any prototypes become available for testing. This is true for railway applications as much as for any other field. Model-based approaches can offer important benefits in terms of cost, safety and timely delivery of final systems in, for example, the development of new train designs and the introduction of improved train operating methods.

Simulation models must always be capable of being used in a convenient and effective fashion for the application at hand. Assessment of the fitness of a given simulation model for a specific application should always include careful testing procedures based on the well-established processes of validation, verification and documentation.

The simulation of longitudinal train motion has attracted new interest recently because of the importance of de-carbonising rail services and the need to reduce energy usage through, for example, the introduction of improved driving strategies. The design of bi-mode traction systems and other hybrid powertrains is one specific area where simulation methods are potentially very important.

In models of longitudinal train motion, the tractive force at any time instant is balanced by an inertial force, plus forces that include aerodynamic and other resistive characteristics of the train, together with route-dependent resistive forces such as gradient resistance, curvature resistance and additional aerodynamic resistance in tunnels.

The objectives of this paper are to review forms of longitudinal train models in current use (e.g. [1]-[7]) and to investigate sources of uncertainty within such models in the context of specific types of application [8]. The use of inverse simulation methods for handling problems involving longitudinal train dynamics is also considered [9].

1 Dynamic Models of Longitudinal Train Motion

The longitudinal motion of a train can be described by a set of nonlinear ordinary differential equations and associated algebraic equations. As in the modelling of other engineering systems, questions of the model structure are linked closely to the intended application. For example, in models used for design of train control systems or the assessment of energy demands, lateral and vertical movements of vehicles are normally neglected and only longitudinal motion is considered.

1.1 General Form of Multi-mass Model

Figure 1 is a schematic diagram of a general lumped-parameter model of the longitudinal motion of a train involving a number of coupled vehicles. Here, the parameters v_1 and k_1 describe viscous and elastic properties of the couplings between the first and second vehicles. Couplings between other vehicles are represented in a similar way.

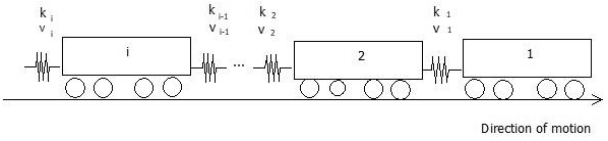


Figure 1: Typical multi-mass representation of a train showing the first second and ith vehicle.

Each powered vehicle is subjected to traction, braking and resistive forces while the unpowered vehicles have no tractive force component. In both cases, train resistance forces involve rolling resistance, air resistance, resistance due to gradients (positive or negative) and resistance due to track curvature.

The equation for the leading vehicle has the form:

$$m_1 \frac{d^2x_1}{dt^2} + v_1\left(\frac{dx_1}{dt} - \frac{dx_2}{dt}\right) + k_1(x_1 - x_2) + F_{R1} \pm F_{G1} + F_{C1} = F_{T1} - F_{B1} \quad (1)$$

where the variables $x_1(t)$ and $x_2(t)$ are the distances relative to the starting point for the first and second vehicles, and the forces F_{R1}, F_{G1} and F_{C1} are resistive, gravitational and track curvature forces.

The forces F_{T1} and F_{B1} are the tractive and braking forces at the rail, respectively. The effective vehicle mass is represented by the parameter m_1 which is the product of the actual static vehicle mass and a factor $(1 + \phi)$ introduced to account for the inertial effects of rotating parts. The parameter ϕ is typically assigned a value of about 0.1 (see e.g. [5]). This equation can be modified to incorporate more complex nonlinear representations of couplings (see, e.g. [1]).

The vehicle resistive force F_{R1} involves three components, as shown in (2). This is traditionally referred to as the Davis equation (and also, the Leitzmann Formel, von Borries Formel or fonction de Berbier) and is based largely on empirical findings. It involves a constant component a_1, a velocity-dependent component involving b_1 and an aerodynamic component with factor c_1 that depends on the square of the velocity.

$$F_{R1}(t) = a_1 + b_1\frac{dx_1}{dt} + c_1\left(\frac{dx_1}{dt}\right)^2 \quad (2)$$

Equations can be derived in a similar way for the other vehicles, giving an equation for the final vehicle, *n*, of the form shown in (3):

$$m_n\frac{d^2n}{dt^2} + v_{n-1}\left(\frac{dx_n}{dt} - \frac{dx_{n-1}}{dt}\right) + k_{n-1}(x_n - x_{n-1}) + F_{Rn} \pm F_{Gn} + F_{Cn} = F_{Tn} - F_{Bn} \quad (3)$$

together with an equation for the resistance F_{Rn}, which is similar in form to (2).

These equations apply to any combination of powered and unpowered vehicles and can therefore be used to describe multiple-unit passenger trains with several powered wheel-sets, or locomotive-hauled passenger or freight trains (e.g. [1], [6], [7]).

This distributed mass model is potentially useful for applications requiring consideration of the kinetic energy in different parts of a train and investigation of associated transient forces that may be exerted at the couplings. Such issues can be important in the modelling and simulation of long trains, especially in the context of train control and braking strategies. This is particularly important for long freight trains, especially on routes with frequent changes of gradient, or many sharp curves and local speed restrictions.

1.2 The Single-mass Model

In many cases where the train length is short compared with features of the route, the model of (1) – (3) may be reduced to involve a single mass (see, e.g. [2]). Comprehensive testing of the validity of this simplified generic model structure has been undertaken for short trains such three-coach or four-coach diesel or electric multiple units, including the use of on-train data (see, e.g. [2], [6], [10]).

Such models have been used widely for investigations of fuel consumption or energy usage.

With this single point-mass approximation, the model of (1) – (3) reduces to:

$$M\frac{d^2x(t)}{dt^2} + R(t) \pm F_G + F_C = F_T(t) - F_B(t) \quad (4)$$

$$\text{where} \quad R(t) = A + B\frac{dx}{dt} + C\left(\frac{dx}{dt}\right)^2 \quad (5)$$

Here the variable $x(t)$ is the position of a chosen reference point on the train (usually the front or the mid-point of the train). The other variables $F_T(t)$, $F_B(t)$ and $R(t)$ are the tractive, braking and train resistance forces acting on the effective mass M_e of the train. This effective train mass is equal to the sum, *M*, of the static masses of the individual vehicles, multiplied again by an inertial factor $(1+\phi)$. The parameters *A*, *B* and *C* are counterparts of the parameters a_i, b_i, and c_i in the train resistance terms of the distributed-mass model.

1.3 The Route Sub-model

The gravitational terms included in (1) and (3) depend on the gradient profile and this information may be taken from a sub-model providing data for each point on the chosen route.

As shown in Figure 2, the gravitational force acting on each vehicle in the multi-mass model is given by:

$$F_{Gi} = m_i g \sin\alpha \tag{6}$$

where $\sin\alpha = \frac{1}{Y}$ for a gradient of 1 in *Y* at each point on the route for vehicle *i* and g is the acceleration due to gravity.

Similarly. for the single-mass model of (4), the angle α gives the gradient at each point on the track and the force acting on the train due to gravity is given by:

$$F_G = Mg \sin\alpha \tag{7}$$

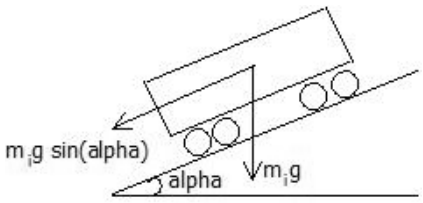


Figure 2: Diagram showing gravitational force acting on the *ith* vehicle of the train.

The route sub-model can also incorporate track curvature data to allow resistance forces $F_{Ci}(x_i(t))$ or $F_C(t)$ to be found.

This is usually based on an empirical approximation and the most widely used relationship involves the product of mass and a factor that depends inversely on curve radius [3] to give:

$$F_{Ci}(x_i(t)) = \frac{k_{cv}}{R(x_i(t))}\, m_i \tag{8}$$

in the case of the multi-mass model, or

$$F_C(x(t)) = \frac{k_{cv}}{R(x(t))} M \tag{9}$$

for the single-mass approximation. Here $R(x_i(t))$ or $R(x(t))$ represents the radius of curvature (m) at the point $x_i(t)$ or $x(t)$ and k_{cv} is an empirical factor that can vary considerably, depending on environ-mental conditions, state of maintenance of the track and condition of the vehicle.

1.4 Tractive Force, Power and Energy

The sum of the forces $F_T(t)$ and $F_B(t)$ forms a net tractive force variable $T(t)$ which can be positive, zero or negative, depending on the operating condi-tion. Note that, in practical simulation models, the braking force component $F_B(t)$ of the tractive force $T(t)$ can only take a non-zero value if $F_T(t)$ is zero.

In the single-mass model, the power at the rail, $P_R(t)$, is given by:

$$P_R(t) = T(t)\frac{dx}{dt} \tag{10}$$

and a similar equation can be used to determine the power at the rail P_{Ri} of any powered vehicle *i* within the distributed mass model.

The applied tractive force is limited at low speeds in order to avoid wheel slip. This commonly involves application of a constant tractive force at the rail, T_0, until the speed reaches a value V_{ch} given by:

$$V_{ch} = \frac{P_R}{T_0} \tag{11}$$

where P_R is the value of the applied power at the rail. This condition corresponds to the point on the hyperbolic curve of (10) where the tractive force at the rail is equal to the limiting value T_0, as shown in Figure 3.

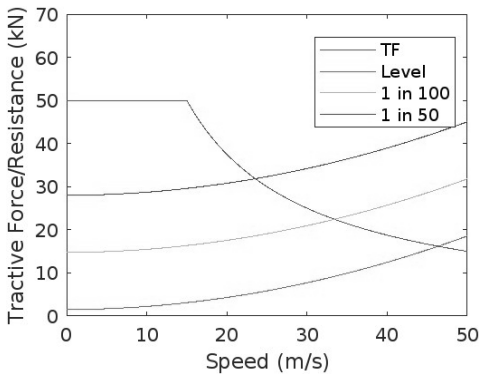


Figure 3: A typical tractive force versus speed curve (TF) together with three train resistance curves for level track (lower curve), a rising gradient of 1 in 100 (middle curve) and a rising gradient of 1 in 50 (upper curve).

It should be noted that this limiting value is mass-dependent and that (10) and (11) apply also to powered vehicles within a multi-mass type of model.

The tractive force and resistance curves of Figure 3 are typical of those for a type of single-mass three-coach multiple unit train used in the United Kingdom. The power level for the hyperbolic part of the curve (defined by (10)) is 750 kW (at the rail). The curves show clearly the variation of resistive forces with speed and also the speed at which the tractive force at the rail and resistance forces balance and how this changes with the gradient.

The energy used for traction is the integral of the power with respect to time, as shown in (12).

$$E = \int P(t)dt \tag{12}$$

Powertrain energy losses and the additional energy for auxiliaries, such as on-board computer systems, heating and air conditioning must also be allowed for.

2 Fitness for Purpose Issues

2.1 Model Structure Issues

For any model, in any field of application, questions concerning model structure are always linked closely to the purpose of the model. A desirable model structure might be thought of as a generic one capable of being used in many different ways, but this approach tends to produce models having a very large number of variables, many of which may not be readily accessed for measurement and also many parameters that are hard to estimate.

For example, the use of a multi-mass description introduces complexities because each vehicle has its own resistance and braking characteristics with inherent uncertainties, and there may also be significant issues about the modelling of inter-vehicle couplings.

Simple linear descriptions of coupling dynamics (such as those used in (1) and (3) above) are unlikely to provide an adequate description of vehicle interactions and more complex nonlinear spring and damper representations may be needed [1]. Often, a single-mass type of structure may be more appropriate since that involves far fewer variables and fewer parametric uncertainties. With the single-mass type of structure the limits within which models can be used with confidence are thus more readily established.

The modelling of train braking systems introduces additional problems and, in many cases, enhancement of the structure of vehicle models becomes essential in order to describe practical systems.

Detailed modelling of braking action is becoming increasingly important due to the current interest in regenerative braking, the development of blended pneumatic/regenerative braking strategies and the introduction of more autonomy within train operations.

In the case of electrified railways, train models may also need to include features of the electrical supply infrastructure to allow investigation of the effects of train dynamics on the overall energy demands and also how supply systems may be affected by regenerative braking. This greatly extends the model boundaries and can add to the complexity of the model. The inclusion of electrical supply system infrastructure within the model also influences the timescales of interest since relatively fast electrical transients may become important.

2.2 Parameter Sensitivity Issues

The structure of the equations (1)-(5) provides useful information about parameter sensitivities. The mass has an obvious and direct effect through the terms involving acceleration and also through the gradient terms as in (6) and (7). The resistance curves of Figure 3 show clearly that the gradient force is the dominant term in the total resistance to motion, except when the train is travelling on level track or on very gentle gradients.

Although the resistance equations are essentially empirical relationships, much work has been done to try to establish a more physical basis for the coefficients *A*, *B* and *C* (see, e.g. [3]-[8]). For example, coefficient *A* is known to depend on the train mass, while *B* is generally believed to depend both on train mass and on the mass of air entering the train for ventilation and cooling. The aerodynamic term *C* depends on the shape of the train. It is directly proportional to the air density, the frontal area of the train and an overall drag coefficient that is the sum of the head and tail drag coefficient plus other drag terms (e.g. [3]-[5]).

In Figure 3 the sensitivity of the resistance to each of the factors *A*, *B* and *C* of (5) may be seen clearly, with the terms involving *B* and *C* becoming more important as the speed increases. The aerodynamic term (i.e. *C*) is clearly most important at high speeds and its components have been investigated in much detail for high-speed train design using both experimental data and the techniques of computational fluid dynamics (CFD). For trains operating in the lower part of the speed range, as shown in Figure 3, the resistance force depends mainly on the train mass through parameter *A*.

The significance of the resistance forces due to the terms involving the *A*, *B* and *C* coefficients compared with the gradient and curvature resistance forces is best seen from a specific example. The data used to generate the resistance curves of Figure 3 demonstrate, for example, that the gradient component of resistance on a rising gradient of 1 in 100 is similar to the sum of all the other resistance forces at 40 m/s for the type of train considered. For the gradient of 1 in 50, the contribution from the gradient term doubles again. In terms of curvature resistance, calculations based on (9) using a typical value for k_{cv} suggest that, for the multiple unit train considered, the curve resistance force is 2751 N for a relatively sharp curve of 200m radius. This is almost twice the resistance force associated with the *A* term of the Davis equation. This resistance is also believed to be much larger (possibly by as much as a factor of two) if the train is moving away from rest while on a curved section of track. Other relevant issues that affect curve resistance include the amount of cant on the curve, the train speed, the rail profile, the design and condition of the vehicles and whether or not rail track lubrication is applied on the curve.

2.3 Uncertainties in Train Parameters

Getting good estimates of the train resistance parameters can present difficulties, even for an existing type of train since these are empirical quantities, which, at present, are usually estimated from full-scale tests using coasting trials [3], [6], [10]. Finding appropriate parameters for individual vehicles in a multi-mass model presents significant difficulties.

Even for the single-mass type of model, accurate estimates are available only for a few specific types of train (see, e.g. [3]-[5]) and values used in practice are often approximate and based on average values for the types of vehicles in question. For example, values often quoted for the resistance parameters for a two-coach Class 156 diesel multiple unit train (a type widely used in the United Kingdom) are A=1500 N, B=6.0 $Nm^{-1}s$ and C=6.7 $Nm^{-2}s^2$ but it is known that these values can vary significantly. For example, with all seats occupied, the value of *A* could, in theory, rise to 1755 and the value of parameter *B* would also be increased.

In the case of new passenger train design, or new freight vehicles, the determination of appropriate values for resistance parameters is clearly difficult in the initial stages of the design and development process.

Once a prototype is available it may be possible to estimate resistance parameters during the initial testing that is often carried out prior to a new train design being approved for service.

2.4 Uncertainties in Route Parameters

Although nominal gradient information is often available, uncertainties may still exist because of the cumulative effects of changes made during track maintenance work and upgrades over long periods of time.

Major changes may have been recorded but the effects of minor adjustments made over many years of track maintenance may not always be known. For example, even changes as small as 25 cm at each end of a 1 km section of track with a nominal gradient of 1 in 100 could lead to an average gradient change to 1 in 95 for that section.

Information about track curvature may also be available but this may be more difficult to incorporate accurately within the route sub-model as, in practice, curved sections of track may involve a transition at the start and finish where the radius changes gradually.

Also, even on a curve having a fixed radius, the parameter k_{cv} can vary between about 500 and 1200 depending on environmental conditions and train speed.

3 Inverse Methods

There are modelling situations in which an inverse approach may allow some issues to be considered directly that would require much repetitive simulation if approached through traditional forward simulation methods. For example, a time history of distance versus time for an existing type of train on a specific route is potentially useful in providing a reference schedule which can be applied as input to an inverse simulation for a new train design.

Estimates can then be obtained directly for the tractive force, power and energy required for that new train operating on that specific route for that schedule. Time-scaling methods allow a reference schedule of this kind to be adjusted to investigate possible performance enhancements [9].

Several methods of inverse simulation are available. One approach that has been used successfully with train models involves a continuous system simulation method based on feedback principles [9].

4 Model Testing

For models developed for system design purposes, no system is available initially to allow rigorous model testing and validation. Only when a prototype system becomes available is it possible to make detailed comparisons between the model are the real system behaviour (e.g. [6], [7], [10]). However, even at an early stage of a new design project, insight may be gained from the use of previously tested models of a broadly similar type and by basing the work on well-established and fully tested models of components. In most design applications repeated testing and refinement of models is essential as the project progresses.

For models of existing trains, data gathered on-board a train (e.g. speed, distance travelled, power or tractive force applied at each time instant over a specific route) can provide quantitative data to allow assessment of predictive accuracy (e.g. [6], [7]). In this context, care in needed in considering use of global positioning system (GPS) data due to inherent errors. Checking of GPS values against independently recorded data based on passing times at key points on the route in question is recommended to detect any inconsistencies.

Some published reports on train performance monitoring suggest that gradients can be estimated from GPS altitude data. However, possible errors in the GPS altitude channel are known to be significantly larger than errors in positional and speed data. Conventional gradient profiles are therefore usually used in train simulations, rather than altitude data.

5 Applications

The choice of model structure depends on the objectives. For example, if the train length is short compared with features of the route, such as the gradient profile and curves, the single-mass type of description may be adequate. However, if the train is longer and especially if it is locomotive-hauled, a multi-mass description may be needed, particularly if the dynamics of different sections of the train are considered important or there is a need to estimate forces at couplings [1]. For example, with regenerative braking alone, the fact that there are no powered axles on trailing vehicles means that forces applied to the locomotive can become unacceptably large during braking actions. However, in conventional pneumatic brake systems, there can be significant delays in the application of the brakes on different vehicles and this can also give rise to undesirable longitudinal dynamic effects.

Thus, investigations of braking strategies and the possible use of blended braking systems involving a combination of pneumatic and regenerative braking, especially with long trains, may require the use of a multi-mass type of model.

In the context of fitness-for-purpose, two specific areas of application have been chosen for discussion because of their current relevance in terms of energy costs and the moves towards de-carbonisation of rail transport. The first of these areas concerns the development of more economical methods of train operation and driving, including lightweight train designs and regenerative braking. The second application area considered involves the design of hybrid trains based on hydrogen fuel cells and batteries for routes where traffic levels do not justify conventional electrification.

5.1 Train Operations Simulation

Dynamic models and simulation methods are already being used to investigate efficiency improvements. Examples of strategies for minimising fuel and energy costs in train operations include the development of improved driving practices and also consideration of possible benefits of infrastructure upgrades on sections of the route where there are severe speed restrictions. As well as reducing travel times, such upgrades could avoid wasting energy through brake applications and subsequent acceleration back to the normal line speed. Other applications of simulation include the investigation of interactions between train operations and the electrical supply system on electrified railways and studies of the use of regenerative braking.

In all such applications, model accuracy is of central importance since inadequate simulation models inevitably lead to incorrect conclusions. For example, optimisation calculations must often be repeated for a range of different model parameter values for different train load conditions.

For passenger services, loads may vary from the minimum (tare weight of the train plus any fuel load and driver) to crush loading where every seat is occupied and there are many standing passengers.

This mass difference is always significant and is particularly large on light rail vehicles. As an example, the CAF Urbos3 trams currently used in Edinburgh have a tare mass of 56000kg with 250 places available. Using an average passenger weight allowance of 80 kg, a fully-loaded tram has a mass of about 76000 kg representing an increase in mass of almost 36% [11].

This is a very significant change that can have a marked effect on longitudinal dynamics and especially performance when accelerating and braking. For a typical UK three-coach diesel multiple-unit (dmu) train, the extra mass contributed by a full load of passengers is smaller, but still significant, at about 15% of the tare value. For crush-loaded conditions, with the maximum possible number of standing and seated passengers, the load increase could be at least 10% greater.

Inverse simulation methods are potentially useful for the investigation of driving strategies and also for the design of braking systems. In the inverse approach, the input variable would be a required profile of distance, speed or acceleration versus time and the simulation model outputs would include the power, tractive force or braking force at the rail.

Again, there are important constraints that must be taken into account in terms of adhesion limits, maximum power levels for motors in generator mode, maximum power levels during the charging of the energy storage system and constraints in terms of the ratings of power-electronic components.

Taking the Edinburgh tram example, it would be interesting to use simulation to investigate whether or not regenerative braking to an on-board supercapacitor system or battery pack could lead to useful operating economies, taking account of weight penalties arising from the additional on-board equipment.

5.2 Hybrid Powertrain Simulations

The current drive to de-carbonise both passenger and freight rail services has generated a strong interest in the development of forms of traction involving combinations of hydrogen fuel cells and electrical storage elements such as batteries, supercapacitors or flywheels. The need for energy storage arises because hydrogen fuel cells tend to be sluggish in their response to changes in demanded power and the stored energy can be used to provide additional tractive effort very rapidly when the train is accelerating or ascending steep gradients. Re-charging takes place when all of the power available from the fuel cell stack is not needed for traction and also during regenerative braking. Further details of hybrid powertrain configurations may be found elsewhere (e.g. [12]-[14])

Supercapacitors (and possibly flywheels) are suitable for light rail vehicles on routes with frequent stops, but batteries are generally considered more appropriate for other applications.

However, batteries introduce significant weight penalties and getting the right balance between the sizes of the fuel-cell stack and battery pack is a complex process. The space required for the storage of hydrogen gas onboard the train is another critically important factor. Simulation methods have a potentially important role in addressing all of these design issues (e.g. [12]-[14]).

The design of hybrid powertrains is also an area in which inverse models are being applied. That approach has been used for automotive powertrain design for some considerable time, but mostly using steady-state or quasi-steady descriptions.

However, in railway applications, the dynamics of the train itself, together with the characteristics of the route, are very important and steady-state descriptions are inappropriate. It is believed that inverse dynamic modelling and simulation can provide important additional insight into the sizing of powertrain components and the optimization of the associated control and energy management systems [12]-[14].

Simulation activities linked to the conversion of a former ScotRail electric multiple-unit train to hydrogen fuel-cell/battery-electric hybrid form are discussed in [13] and further simulation results may be found elsewhere (e.g. [14]). The routes concerned are typical of lines that provide important transport links but involve traffic densities that are too low for a strong business case for conventional electrification. In general, it has been found that hillier routes lead to powertrain configurations involving larger batteries to cope with the frequent changes in demanded power level.

Figures 4(a) – 4(d) show typical inverse simulation results for a specific hybrid train configuration involving hydrogen fuel-cell/battery-electric traction.

This is for a route section approximately 15 km long, which is typical of distances between stations on some rural routes in Scotland. The gradient profile involves an initial section of level track for 1 km where the train accelerates towards the maximum permitted speed of 96 km/hr, a 4 km section with a rising gradient of 1 in 55 and a final section where the train operates on level track, including a coasting phase and a final braking phase. These features of the route are reflected in the speed time-history of Figure 4(b) which corresponds to the distance versus time schedule in Figure 4(a) (generated using a simulation of a conventional diesel multiple unit).

This schedule provides the input to the inverse simulation and Figure 4(c) shows the tractive force that needs to be developed by the hybrid train to match the distance versus time record of Figure 4(a). Negative values of tractive force correspond to braking actions and Figure 4(d) shows the energy usage, including energy recovered through regenerative braking. Such results then provide a basis for decisions regarding the power rating of the fuel-cell stack and battery pack as discussed in [13] and [14]. Results in Figures 4(a) – 4(d) are for a set of parameters corresponding to conditions involving a full load of seated passengers.

This use of the inverse simulation approach, as illustrated in Figures 4(a) – 4(d), allows the necessary total tractive force, total power and total energy to be found for the new train for the set of performance requirements defined using the simulation of the existing dmu. Some of the necessary power is provided by the fuel-cell stack directly and some from energy stored in the battery pack. Results from the inverse simulation provide a starting point for investigation the optimum sizes of these components to ensure that the necessary total power is available at all times, while also satisfying constraints such as limits on the allowable battery state of charge and power ratings of electronic converters. This information then provides a basis for the detailed design of the powertrain control and energy management systems.

Variation of parameters of the train such as *M*, *A*, *B* and *C* or of the route sub-model, such as the gradient profile and curve resistance values, affect the tractive force and energy records in ways that are consistent with results of parameter sensitivity analysis. For example, reducing the mass to correspond to the tare condition with no passengers on board reduces the peak energy value shown in Figure 4(d) by about 10%. Similarly, simulation results suggest that by doubling the passenger load the peak energy use is increased by about 10%.

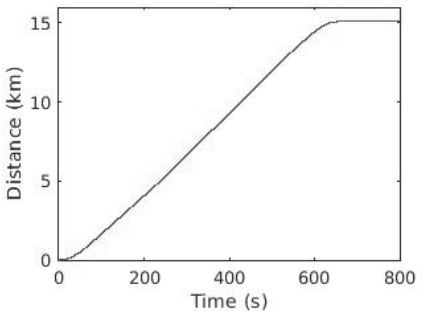


Figure 4(a): Distance versus time reference schedule applied as input to the inverse simulation.

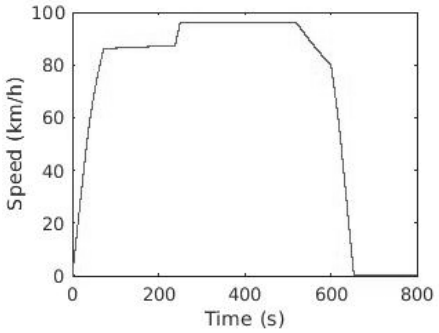


Figure 4(b): Speed versus time record corresponding to the reference schedule of Figure 4(a).

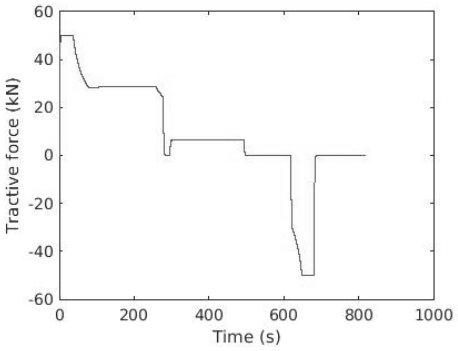


Figure 4(c): Tractive force time-history for a proposed three-coach hybrid multiple-unit train obtained using inverse simulation (for distance versus time schedule of Figure 4(a)).

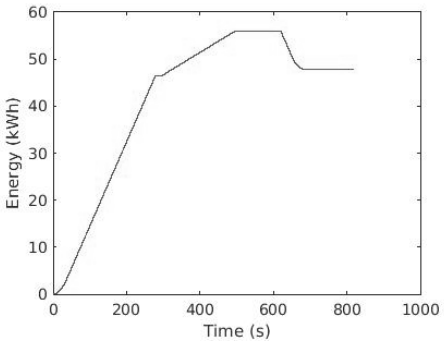


Figure 4(d): Energy record from the inverse simulation for the hybrid train for same conditions as in Figures 4(a), 4(b) and 4(c).

The model used in this work includes a simple representation of driver action in which speed is compared continuously with the speed limit for the current train position. Using that difference, tractive force values at each time step in the simulation are multiplied by a factor representing driver control actions in approaching a limit and adhering to it [9]. Control actions associated with the start and end of coasting and the initiation of braking are introduced through the route model.

6 Discussion

Uncertainties within dynamic models of the longitudinal motion of trains tend to be larger than in many other engineering applications and it is essential to record all inherent assumptions and approximations.

Users must have a good understanding of the range of values possible for each parameter within the chosen model structure and must make full use of this information when applying the model.

Rigorous model testing procedures are essential in ensuring fitness for purpose. The need for repeated testing whenever model changes are made requires robust model management processes, including formal procedures for model version control and the updating of documentation. Ideally, a simulation model needs to be maintained throughout the lifetime of the system that it represents. Not only does this ensure that a model remains fit for its original purpose but it also means that a model developed during one design project may provide a useful starting point for models in future projects. This raises some important issues about the use of simulation models in railway applications that may not, at present, be fully recognised within the industry.

7 Conclusions

Models used to represent the longitudinal dynamics of a single train have many potential applications, including the investigation of more efficient operating strategies and the design of new trains. The fitness for purpose of any model depends on decisions about its boundaries, the extent to which it satisfies accuracy requirements and whether it is to be used in a conventional forward fashion or using an inverse approach.

The model structure, boundaries, parameters and timescales must, therefore, all be tailored to the intended use. Often, a description involving a single-mass approximation is a convenient representation for short multiple-unit passenger trains.

However, multi-mass models may be preferred for longer locomotive-hauled passenger or freight trains, despite the additional complexities. In both the multi-mass and single-mass types of description there are a number of key parameters that have to be chosen to describe the resistance characteristics of the train.

Accurate information must also be available about the gradient profile and track curvature changes for the route in question.

Moves towards de-carbonisation of rail transport and the development of unconventional forms of traction, such as hybrid trains involving fuel-cell/battery-electric powertrains, often lead to complex design problems where the use of dynamic simulation methods can provide valuable insight. However, parametric and structural uncertainties mean that using an envelope of simulation results rather than time histories for a single set of parameter values can be very important.

Dynamic models and computer simulation tools have an increasingly important role in many other railway applications beyond those reviewed in this paper. Fitness-for-purpose is an issue of central importance in all of these and, as in other application areas, all model uncertainties and limitations must be identified, recorded in model documentation, and acknowledged by potential users.

References

[1] Cole C. Longitudinal train dynamics. In: Inwnicki, S, editor. *Handbook of Railway Vehicle Dynamics*, 1st edition. USA: Boca Raton, CRC; 2006. p 40.

[2] Lu S, Hillmansen S, Ho M, Roberts C. Single train trajectory optimisation. *IEEE Trans. Intell. Transp. Syst.* 2013; 14(2): 743-750. DOI 10.1109/TITS.2012.2234118

[3] Rochard B P, Schmid F. A review of methods to measure and calculate train resistances, *Proc. IMechE, Part F: J. Rail & Rapid Transit.* 2000; 214(4): 185-199. DOI 10.1243/0954409001531306.

[4] Armstrong P H, Swift D S. *Low energy Part A: Identification of energy use in multiple units. Report MR VS 077.* UK: British Rail Research, Derby; 1990.

[5] Anonymous. *Quantification of benefits of train mass reduction, Annexes E*: UK: RSSB (Rail Safety and Standards Board) , London; May 2010,

[6] Lukaszwicz P. A simple method to determine train running resistance from full-scale measurements, *Proc. IMechE, Part F: J. Rail & Rapid Transit.* 2007; 221(3): 331-337. DOI 10.1243/09544097JRRT88

[7] Rangelov V N. Gradient modelling with calibrated train performance models. In Brebbia CA, Tomi N, Mera J M et al., editors. *Computers in Railways XIII (WIT Trans on Built Environment, 127).* UK, Southmapton: WIT, 2012. p 11.

[8] Boschetti G, Mariscotti A. The parameters of motion mechanical equations as a source of uncertainty for traction systems simulation. In *Proceedings, 20th IMEKO World Congress*, 2012 Sept; Busan, Rep of Korea. Budapest, Hungary: IMEKO, 2012. 897-902. DOI 10.5281/zenodo.3604806

[9] Murray-Smith D J. Development of an inverse simulation method for the analysis of train performance, *Proc. IMechE, Part F: J. Rail & Rapid Transit.* 2018; 232(5): 1295-1308.
DOI 10.1177/0954409717720349

[10] Aradi S, Bécsi T, Gáspár P. Estimation of running resistance of electric trains based on on-board telematics systems, *Int. J. Heavy Vehicle Systems.* 2015; 22(3): 277-291.
DOI 10.1504/IJHVS.2015.071689

[11] Tian Z, Zhao N, Hillmansen S, Roberts C, Dowens T, Kerr C. SmartDrive: Traction energy optimization and application in rail systems, *IEEE Trans. Intell. Transp. Syst.* 2019; 20(7): 2764-2773.
DOI 10.1109/TITS.2019.2897279

[12] Pugi L, Berzi L, Cirillo F et al A tool for rapid simulation and sizing of hybrid traction systems with fuel cells *Proc. IMechE, Part F: J. Rail & Rapid Transit.* Online: April 2022.
DOI 10.1177/09544097221092622

[13] Murray-Smith D J, Fisher P. *Hybrid trains for the Highlands? Computer simulations of fuel-cell/battery-electric trains on secondary routes in Scotland.* Report, School of Engineering, University of Glasgow; 2021. p 10.
DOI 10.13140/RG.2.2.32865.43366

[14] Murray-Smith D J, *Modelling and simulation of hybrid electric trains powered by hydrogen fuel cells and batteries for routes in the highlands of Scotland: Results for the case of a three-coach multiple unit train.* Report, School of Engineering, University of Glasgow; 2020. p 23.
DOI 10.13140/RG.2.2.18028.39041

An Open Source Tool for Calculating CO_2 Pipeline Decompression Wave Speed

Anders Andreasen[1*], Leandro-Henrique Sousa[2], Geir Agustsson[2]

[1]Ramboll Energy, Energy Transition, Process Department, Bavnehøjvej 5, 6700 Esbjerg, Denmark;
[*]*anra@ramboll.com*

[2] Ramboll Energy, Energy Transition, Pipelines, Hannemanns Alllé 53, 2300 Copenhagen S, Denmark

SNE 32(4), 2022, 187-193, DOI: 10.11128/sne.32.tn.10622
Received: 2022-06-21; Revised: 2022-08-31
Accepted: 2022-09-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. This paper describes a simplified calculation method for pipeline decompression wave speed based on a rigorous equation of state for pure CO_2 as well as mixtures with significant impurities. Calculations are performed assuming homogeneous equilibrium for the estimation of the speed of sound in the two-phase region and calculations are performed along an isentropic decompression path. These calculations are important for the design of pipelines and can be used to estimate the required wall thickness and/or material toughness when combined with e.g. the Battelle two curve method, thereby ensuring that a potential running ductile fracture is arrested. The calculations are validated against available literature data and is offered as an open source tool. For pure CO_2 at supercritical conditions the model results match experimental results very well, whereas for the dense liquid phase the pressure plateau in the decompression wave speed curve is over-predicted. For CO_2 with impurities the model calculations generally match experimental data, except for the experiment with a significant fraction of hydrogen and for the experiment with the highest amount of impurities of approx. 6%. In these two cases the pressure plateau is under-predicted.

Introduction

Increased focus on carbon dioxide emission reduction is bringing to the forefront several required technologies that can support such a scheme. This includes carbon dioxide capture [1] and transport in various forms, such as in trucks, ships and pipelines, that brings the carbon dioxide from source towards disposal / storage or reuse. For certain conditions of volumes and distances, pipelines are an economic method to transport gasses and liquids [2], as the oil and gas industry has recognized for decades. Pressurized applications allow for a significant increase in transportable volumes. For carbon dioxide at ambient temperatures, this means that transport in so-called dense phase or super critical state is often realised.

Compared to natural gas pipelines a running ductile fracture is of bigger concern for carbon dioxide pipelines. Sometimes the wall thickness dictated by the design pressure is not enough to ensure that a ductile fracture is arrested. In case of a pipeline fracture the fluid decompresses, but when the CO_2 reaches saturated conditions the decompression speed drops significantly. In case the decompression speed is lower than the speed of a running fracture, the pipeline itself cannot arrest a running fracture. In order to properly design against a running ductile fracture it is essential to be able to predict the decompression speed.

By using the Battelle two-curve method (BTCM) [3] the fracture and decompression velocity can be compared by plotting the fracture velocity and decompression velocity as a function of the pressure at the crack tip cf. Figure 1 where the decompression wave speed is calculated for pure CO_2. If the fracture velocity exceeds that of the decompression wave speed the fracture will not be arrested by the pipeline material itself. This is illustrated by the red curve illustrating the fracture velocity of a pipeline either with inadequate wall thickness (or toughness). By increasing the wall thickness it can be ensured that the fracture velocity always stays below the decompression velocity and a fracture will be arrested.

Various tools have been presented in the literature for simple calculations of the decompression wave speed, still following DNV guidelines [5], such as

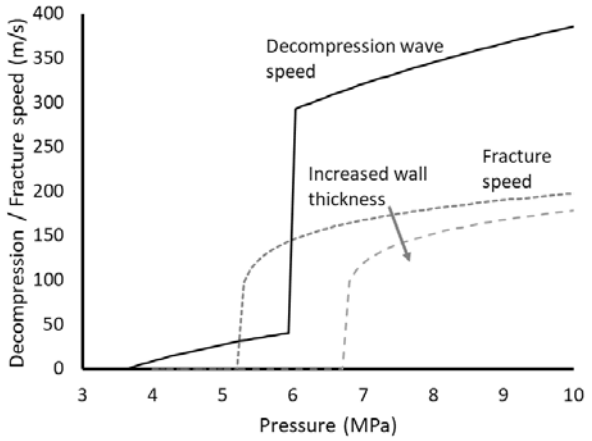


Figure 1: The Battelle two-curve method. The graph has been calculated according to the formulas shown in [4] but only for illustrative purpose.

GASDECOM [6], EPDECOM [7] and DECOM [8], which employ the assumption of unidimensional isentropic, homogeneous equilibrium and inviscid formulation of the decompression wave without explicitly solving the fluid transport equations [9]. On the other end of the scale various 1-D/2-D CFD based tools have also been demonstrated where the mass, momentum, and energy balances are explicitly solved [10, 11, 12, 13]. Common to all tools is that some are purely academic and some have been developed into commercial products. However, none of the aforementioned tools are freely available to the public. In that respect, the tool presented in the present paper differentiates itself from these tools: it is open source and freely available for use by the public.

1 Methods

1.1 Decompression wave speed

In this section the RAMDECOM (RAMboll DECOMpression wave speed) tool is described. In RAMDECOM the calculation methodology for the decompression wave speed follows that shown in [4, 14]. When assuming the decompression wave to be isentropic, in homogeneous equilibrium and inviscid, the decompression wave speed, W, is expressed as:

$$W = C - U \tag{1}$$

where C is the fluid speed of sound and U is the fluid outflow velocity.

The outflow velocity is given at any pressure, P, by:

$$U = -\int_{P_0}^{P} \frac{C d\rho}{\rho} = -\int_{P_i}^{P} \frac{dP}{C\rho} \tag{2}$$

where P_0 is the initial pressure and ρ is the fluid density. Integration is performed along an isentropic path. The outflow velocity in the above equation can be expressed by numerical integration using finite difference:

$$U_i = U_{i-1} + \frac{P_{i-1} - P_i}{C_i \rho_i} \tag{3}$$

where the subscript i refers to the current integration step and $i-1$. Properties from the previous step is known, only density and speed of sound needs evaluation at the new step.

In order to calculate the density and the speed of sound, as well as VLE behaviour, an adequate equation of state is required. In the currect work CoolProp [15] or REFPROP [16] is used as the thermodynamic backend. Both tools use Helmholtz energy formulations for fluids modelling both for pure fluids and for mixtures. For pure CO_2 the Span-Wagner equation of state is employed [17], for mixtures the method of Lemmon [18] and Kunz [19] is used. For mixtures with CO_2 the binary parameters in both CoolProp and REFPROP have been updated with those from EOS-CG [20] and later estimations by Herrig [21].

While the speed of sound is well defined for a single phase fluid, further assumptions are required in order to define it for two-phase / multi-phase. Assuming homogeneous equilibrium the speed of sound can generally be defined as:

$$C = \sqrt{\left(\frac{dP}{d\rho}\right)_s} \approx \sqrt{\left(\frac{P_{i-1} - P_i}{\rho_{i-1} - \rho_i}\right)} \tag{4}$$

where the differential is evaluated at isentropic conditions. The full calculational workflow is the following:

- Define initial conditions: Temperature,T_0, pressure, P_0 and composition (either pure CO_2 or mixture with impurities)
- Calculate density and specific entropy using the equation of state
- For each integration step from the initial pressure the new pressure P_i is set as P_{i-1} minus 1e5 Pa and the new density is evaluated via an isentropic path.

- The speed of sound, C_i, is calculated via Equation 4 assuming a small ΔP of 100 Pa.
- C_i is used in Equation 3 to calculate the outflow velocity U_i
- W_i is calculated using Equation 1

The calculations are generally continued until the calculated decompression wave speed becomes zero or negative. The evaluation of properties and estimation of speed of sound is performed at specified pressure and entropy (equal to the initial entropy) i.e. a PS-problem.

For all calculations the CoolProp python wrapper is used. When using REFPROP as backend this is done still via the CoolProp wrapper. The CoolProp backend is applied only for pure CO_2, since the two-phase mixtures failed to solve in many cases. REFPROP can be specified to be used both for pure CO_2 and for mixtures. This work is a continuation of a previous work [22] with the purpose of building useful engineering tools on top of high quality open source software packages. RAMDECOM is developed entirely in python 3 and also relies on other python packages such as pandas [23], matplotlib [24], and numpy [25].

1.2 Experimental

In order to compare the decompression calculation methodology presented in the previous section with experimental data, various relevant experiments have been sourced from the literature. For decompression of pure CO_2 experiments made by Munkejord *et al.* [26] and Botros *et al.* [27]. For CO_2 rich mixtures the experiments from Botros *et al.* [28] have been sourced.

All the experiments sourced have similar set-up and many things in common. The experiments are performed in a horizontal shock-tube comprised of a number of tubular sections flanged together and equipped with pressure and temperature transducers located along the length of the shock-tube. One end is closed and the other end is equipped with a rupture disc. In order to ensure controllable and uniform temperature the shock-tube is heat-traced and insulated. The facilities have mixing and compression units in order to fill the shock tube with the desired mixture and to the desired initial pressure. For additional information about experimental methods and facility description and further details please refer to the original papers [26, 27].

The experimental test conditions for the pure CO_2 experiments are summarised in Table 1 and the experimental test conditions for the CO_2 rich mixtures are summarised in Table 2.

Exp No.	P (bar)	T (°C)	Source
3	40.4	10.2	Munkejord et al.
6	104	40	Munkejord et al.
8	122.2	24.6	Munkejord et al.
15	340.4	36.5	Botros et al.
31	111.11	35.04	Botros et al.
32A	112.7	8.74	Botros et al.

Table 1: Experimental initial conditions for decompression experiments with pure CO_2 from Munkejord *et al.* [26] and Botros *et al.* [27].

2 Results and Discussion

2.1 Pure CO_2

The experiments for pure CO_2 summarised in Table 1 have all been simulated using the Span & Wagner equation of state as provided by CoolProp. The isentropic decompression path for all simulated cases is shown in Figure 2.The path is from the inital pressure and temperature in the single phase region to the saturation line. The experimental initial conditions cover both gas, liquid (dense phase / supercritical liquid), and supercritical fluid. Once the decompression state reaches the saturation line, the isentrope follows the saturation line with varying phase split. Calculations have been done with the REFPROP backend as well and identical results were obtained (not shown).

Decompression wave speed plots are made for all cases and corresponding experimental data has been sourced from [26, 27]. It shall be noted that the experimental points have been read manually with the aid of the ScanIT program from Amsterchem (`https://www.amsterchem.com/scanit.html`). Some slight inaccuracy during the digitization of data from the original references must be expected and further in some areas the original data was too dense to allow all data points to be extracted. That being said the overall characteristics and the shape of the decompression curves have been retained.

The decompression curves for experiments 6 and 8 from [26] and experiments 31 and 32A from [27] are grouped in the same plot cf. Figure 3 whereas experiments 3 from [26] and 15 from [27] are plotted individually in Figures 4 and 5 , respectively.

			Composition (mole %)							
Exp No.	**Pressure (bar)**	**Temperature (°C)**	**CO_2**	**N_2**	**O_2**	**He**	**Ar**	**CO**	**H_2**	**CH_4**
2	148.3	35.9	94.03	5.82	0.127	0.025				
4	145.6	35.1	96.67		3.33					
7	147.8	36.3	96.52			0.0138				3.47
10B	149.3	35.3	96.77						3.23	
5	144.9	35.6	96.77	0.0025				3.23		
9	154.6	35.2	96.14				3.86			

Table 2: Experimental initial conditions for decompression experiments with impure CO_2 from Botros *et al.* [28].

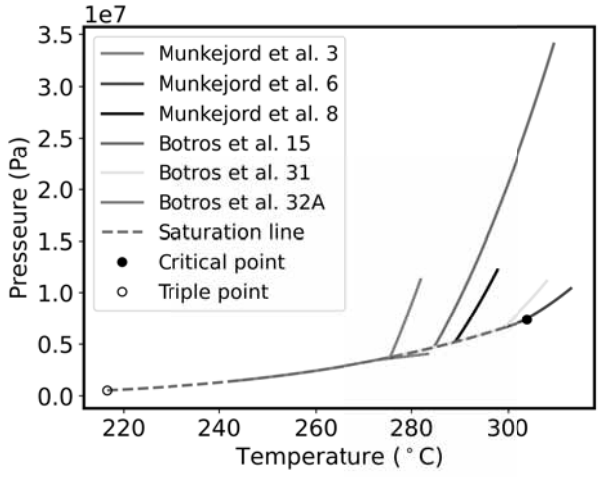


Figure 2: Isentropic decompresion path for all simulated cases for pure CO_2 shown in the P,T plane along with the saturation curve for pure CO_2 from triple point to the critical point.

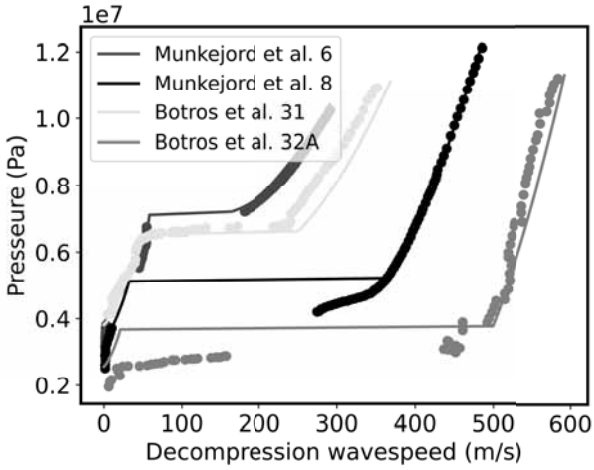


Figure 3: Decompression curves for pure CO_2 calculated with the RAMDECOM code and experimental data.

Generally, the following observations are made: First of all, the isentropic decompression path follows a gradual decrease in decompression wave speed as the pressure is reduced towards the saturation line. Once the saturation line / two-phase state is reached the decompression wave speed abruptly drops due to an abrupt drop in the speed of sound, which is seen as a plateau in the pressure. Second, it seems that the described decompression wave speed model matches experiments very well for decompression from a supercritical fluid state and this applies to experiments no. 6 [26], 15 [27] and 31 [27] as seen from Figure 3 and 5. Especially, the plateau pressure as described previously, is predicted very well for these cases. Finally, the cases where the decompression starts in the supercritical liquid state and in the gas state are predicted less accurately. This applies to experiments no. 8 [26], 32A [27] cf. Figure 3 and to some extent experiment no. 3 [26] cf. Figure 4.

In case of experiment no. 3 in Figure 4, the model predicts a slight pressure plateau at around 3.5 MPa, which is where the theoretical isentropic path intersects the saturation line. However, the experimental data shows a plateau around 2.8 MPa, somewhat lower. Munkejord *et al.* [26] demonstrated that the experimental data was reasonably described as if a single phase isentropic path was followed all the way from the initial pressure to the experimental plateau, also supporting a hypothesis that full equilibrium is not established instantaneously and a significant sub-cooling of the gas phase occurs before the first liquid droplets starts to form.

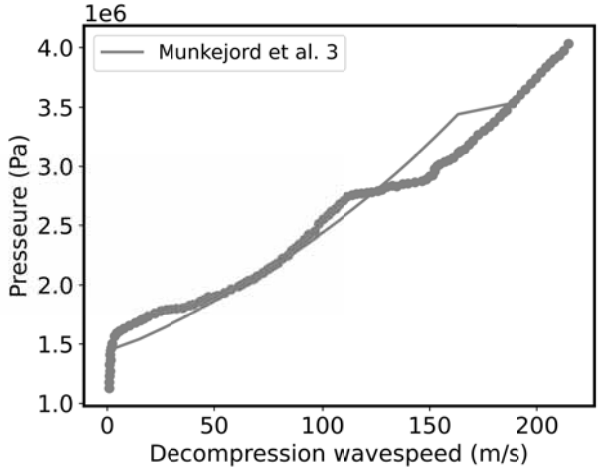


Figure 4: Decompression curve for pure CO_2 calculated with the RAMDECOM code and data for experiment 3 from [26].

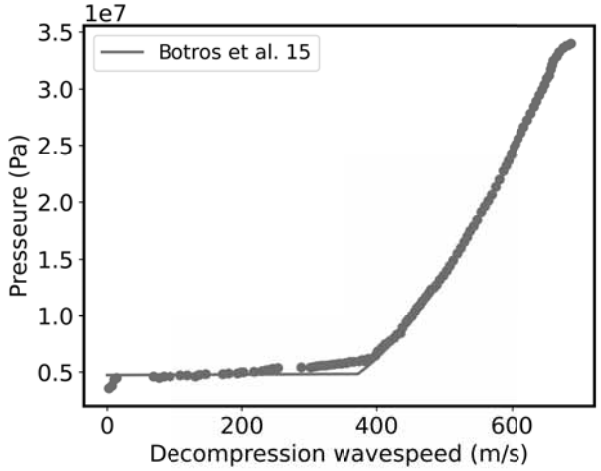


Figure 5: Decompression curve for pure CO_2 calculated with the RAMDECOM code and data for experiment 15 from [27].

Experiments 8 and 32A are very similar and in both cases the discrepancy between the predicted plateau pressure and the experimental data have been rationalised by both Botros *et al.* [27] and Munkejord *et al.* [26] by a very rapid decompression, since the initial pressure is not far from the saturation pressure, in which equilibrium is not reached due to delayed nucleation. This leads to a measured plateau below the predicted. In both this case, and the one for experiment 3 starting from the gas phase, it is evident that any non-equilibrium effects be it delayed nucleation or sub-cooled gas, leads to a conservative result from the simple decompression model.

2.2 CO_2 mixtures

The calculated decompression curves for the CO_2 mixtures in Table 2 are shown in Figure 6 along with the corresponding experimental data from [28]. As seen from the figure experiments no. 4, no. 5, no. 7, and no. 9 are predicted very well by the simple decompression model. The predictions for these experiments are generally in line with both predictions like the one in the present study using the GERG-2008 [28] equation of state as well as predictions made with GASDECOM [6].

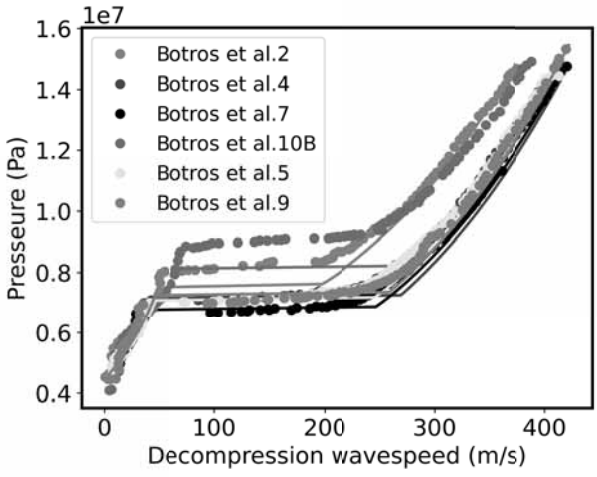


Figure 6: Decompression curves for rich CO_2 mixtures calculated with the RAMDECOM code and experimental data from [28].

The main discrepancies between model and experiment are observed for experiment no. 2 and no. 10B. In both cases the simple decompression model underpredicts the plateau pressure. For 10B, which contains a significant amount of hydrogen in a binary mixture, the same model behaviour is observed by Botros *emph* [28]. As observed for the pure CO_2, the failure to produce equilibrium conditions during experiments generally resulted in an over-prediction of the experimentally observed plateau. The fact that the plateau is underpredicted for the H_2/CO_2 binary mixture could indicate that this is due to a deficiency in the applied equation of state to accurately model the bubble point line in particular. This should be investigated in more detail in future works.

The case of experiment no. 2 is more peculiar. The proposed model would actually be expected to be able to explain the experimental data quite well. Both N_2 and O_2 are main components in combustion gas which EOS-CG targets [20, 21]. Botros *et al.* found good agreement between both GASDECOM [6] and a similar model employing the GERG-2008 equation of state [19] and data for experiment no.2. In the present study the model using the REFPROP back-end under-predicts the plateau pressure by approx. 5 bar.

3 Conclusion

In this paper an open-source tool for calculating the pipeline decompression waves peed for pure CO_2 as well as rich CO_2 mixtures containing significant impurities, using a simplified method, is presented. The tool relies on the Span & Wagner Helmholtz energy equation of state as provided by the open-source tool CoolProp. For mixtures a license for the NIST software REFPROP is required in the present version of the tool.

The calculations have been compared with available experiments from the literature, generally showing good agreement for most of the investigated cases, both for pure CO_2 as well as mixtures. For pure CO_2 the comparisons with experiments reveal that for dense phase / super-critical liquid with initial pressures moderately above the critical pressure, there is a tendency for the pressure plateau to be overestimated, apparently due to non-equilibrium effects. The same applies when decompression is made from an initial gas phase below the critical point. In both cases the inadequacies of the model is to the conservative side when considering fracture behaviour. For CO_2 mixtures the majority of the experimental test cases were predicted very accurately, except for the mixture containing hydrogen and for the mixture with highest level of impurities (approx. 6 mole %). For those two cases the results obtained were non-conservative i.e. the experimental pressure plateau was underestimated.

Acknowledgement

Language secretary Susanne Tolstrup, Ramboll Energy Transition, Process and Technical Safety, is greatly acknowledged for proofreading the present manuscript. The contents of the present paper has been highly motivated by Ramboll's participation in the INNO-CCUS partnership.

The INNO-CCUS Partnership is established as a mean to secure a significant contribution to achieve the Danish government's climate goals on CO_2 reduction, through CCUS solutions.

References

[1] Andreasen, A. Optimisation of carbon capture from flue gas from a Waste-to-Energy plant using surrogate modelling and global optimisation. *Oil Gas Sci Technol - Rev IFP Energies nouvelles*. 2021;76:55.

[2] Lu H, Ma X, Huang K, Fu L, Azimi M. Carbon dioxide transport via pipelines: A systematic review. *Journal of Cleaner Production*. 2020;266:121994.

[3] Eiber RJ, Maxey WA, Bubenik TA. *Final report on fracture control technology for natural gas pipelines*. Columbus, Ohio: American Gas Association., Pipeline Research Committee. Line Pipe Research Supervisory Committee, Battelle. 1993.

[4] Hu Q, Zhang N, Li Y, Wang W, Zhu J, Gong J. A New Model for Calculation of Arrest Toughness in the Fracture Process of the Supercritical CO_2 Pipeline. *ACS Omega*. 2021;6(26):16804–16815.

[5] DNV. *Design and operation of carbon dioxide pipelines - Recommended practice DNV-RP-F104*, February 2021 ed.

[6] Cosham A, Eiber RJ, Clark EB. GASDECOM: Carbon Dioxide and Other Components. vol. 2 of *2010 8th International Pipeline Conference*. 2010; pp. 777–794.

[7] Lu C, Michal G, Elshahomi A, Godbole A, Venton P, Botros KK, Fletcher L, Rothwell B. Investigation of the effects of pipe wall roughness and pipe diameter on the decompression wave speed in natural gas pipelines. *Proceedings of the Biennial International Pipeline Conference, IPC*. 2012;3:315–322.

[8] Cosham A, Jones DG, Armstrong K, Allason D, Barnett J. The Decompression Behaviour of Carbon Dioxide in the Dense Phase. vol. 3: Materials and Joining of *International Pipeline Conference*. 2012; pp. 447–464.

[9] Liu B, Liu X, Lu C, Godbole A, Michal G, Tieu AK. Multi-phase decompression modeling of CO_2 pipelines. *Greenhouse Gases: Science and Technology*. 2017; 7(4):665–679.

[10] Xu BP, Jie HE, Wen JX. A pipeline depressurization model for fast decompression and slow blowdown. *International Journal of Pressure Vessels and Piping*. 2014;123-124:60–69.

[11] Oke A, Mahgerefteh H, Economou I, Rykov Y. A transient outflow model for pipeline puncture. *Chemical Engineering Science*. 2003;58(20):4591–4604.

[12] Elshahomi A, Lu C, Michal G, Liu X, Godbole A, Venton P. Decompression wave speed in CO_2 mixtures: CFD modelling with the GERG-2008 equation of state. *Applied Energy*. 2015;140:20–32.

[13] Fang Y, Poncet S, Nesreddine H, Bartosiewicz Y. An open-source density-based solver for two-phase CO_2 compressible flows: Verification and validation. *International Journal of Refrigeration*. 2019; 106:526–538.

[14] Gu S, Li Y, Teng L, Hu Q, Zhang D, Ye X, Wang C, Wang J, Iglauer S. A new model for predicting the decompression behavior of CO_2 mixtures in various phases. *Process Safety and Environmental Protection*. 2018;120:237–247.

[15] Bell IH, Wronski J, Quoilin S, Lemort V. Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp. *Industrial & Engineering Chemistry Research*. 2014;53(6):2498–2508.

[16] Lemmon EW, Bell IH, Huber ML, McLinden MO. NIST Standard Reference Database 23: Reference Fluid Thermodynamic and Transport Properties-REFPROP, Version 10.0, National Institute of Standards and Technology. 2018.
URL `https://www.nist.gov/srd/refprop`

[17] Span R, Wagner W. A New Equation of State for Carbon Dioxide Covering the Fluid Region from the Triple-Point Temperature to 1100 K at Pressures up to 800 MPa. *Journal of Physical and Chemical Reference Data*. 2009;25(6):1509.
URL `https://aip.scitation.org/doi/abs/10.1063/1.555991`

[18] Lemmon EW, Jacobsen RT. A Generalized Model for the Thermodynamic Properties of Mixtures. *International Journal of Thermophysics 1999 20:3*. 1999;20(3):825–835.

[19] Kunz O, Wagner W. The GERG-2008 Wide-Range Equation of State for Natural Gases and Other Mixtures: An Expansion of GERG-2004. *Journal of Chemical and Engineering Data*. 2012;57(11):3032–3091.

[20] Gernert J, Span R. EOS–CG: A Helmholtz energy mixture model for humid gases and CCS mixtures. *The Journal of Chemical Thermodynamics*. 2016; 93:274–293.

[21] Herrig S. New Helmholtz-Energy Equations of State for Pure Fluids and CCS-Relevant Mixtures. Ph.D. thesis, Ruhr Universität Bochum, Bochum. 2018.

[22] Andreasen A. HydDown: A Python package for calculation of hydrogen (or other gas) pressure vessel filling and discharge. *Journal of Open Source Software*. 2021;6(66):3695.

[23] McKinney W. Data Structures for Statistical Computing in Python. In: *Proceedings of the 9th Python in Science Conference*, edited by van der Walt S, Millman J. 2010; pp. 51 – 56.

[24] Hunter JD. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*. 2007;9(3):90–95.

[25] Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, Wieser E, Taylor J, Berg S, Smith NJ, Kern R, Picus M, Hoyer S, van Kerkwijk MH, Brett M, Haldane A, del R'ıo JF, Wiebe M, Peterson P, G'erard-Marchant P, Sheppard K, Reddy T, Weckesser W, Abbasi H, Gohlke C, Oliphant TE. Array programming with NumPy. *Nature*. 2020; 585(7825):357–362.

[26] Munkejord ST, Austegard A, Deng H, Hammer M, Stang HJ, Løvseth SW. Depressurization of CO_2 in a pipe: High-resolution pressure and temperature data and comparison with model predictions. *Energy*. 2020; 211:118560.

[27] Botros KK, Geerligs J, Rothwell B, Robinson T. Measurements of Decompression Wave Speed in Pure Carbon Dioxide and Comparison With Predictions by Equation of State. *Journal of Pressure Vessel Technology*. 2015;138(3).

[28] Botros KK, Geerligs J, Rothwell B, Robinson T. Measurements of Decompression Wave Speed in Binary Mixtures of Carbon Dioxide and Impurities. *Journal of Pressure Vessel Technology*. 2016;139(2).

Appendix

The code of the calculations described in the present paper is available from the following GitHub repository: `https://github.com/andr1976/ramdecom` including all data and scripts used for preparing the results presented. An example application is also available as a streamlit app at `https://share.streamlit.io/andr1976/ramdecom/main/scripts/streamlit_app.py`, where decompression speed calculations can be made for pure CO_2 at varying initial conditions. The results can be plotted and spreadsheet data can be saved locally.

Implementing Standard Examples with NSA-DEVS

David Jammer[1,2], Peter Junglas[2*], Thorsten Pawletta[1], Sven Pawletta[1]

[1]Research Group Computational Engineering and Automation, University of Applied Sciences Wismar, Philipp-Müller-Straße 4, 23966 Wismar, Germany; *peter@peter-junglas.de*

[2]PHWT-Institut, PHWT Vechta/Diepholz, Am Campus 2, 49356 Diepholz, Germany;

SNE 32(4), 2022, 195-202, DOI: 10.11128/sne.32.tn.10623
Received: 2022-09-27; Revised: 2022-11-07
Accepted: 2022-11-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. To utilize the PDEVS formalism for the practical modeling and simulation of discrete-event systems, the recently proposed variant NSA-DEVS combines the Mealy behaviour of RPDEVS with a simple simulator algorithm by employing infinitesimal time delays. To further test the practical usefulness of this new approach, four simple systems showing non-trivial event-cascades are modeled and simulated within a concrete NSA-DEVS environment: A comparator-switch model, a digital circuit with flip-flops, a basic queue-server system and a more complex queuing system. Their simple implementations show that the potentially large number of delay parameters in NSA-DEVS in practice reduces to a single default value, which only occasionally has to be tuned to adapt to complex causal behaviour. In addition, by providing precise formal definitions of the models and by looking closely at the behaviour of the abstract simulator, the validity of the NSA-DEVS formalism is further substantiated.

Introduction

The PDEVS formalism [1] is a well-established method to concisely describe the hierarchical composition and dynamic behaviour of discrete-event based models. To make it directly applicable for concrete modeling and simulation environments, several variations have been proposed [2] ranging from the introduction of input and output ports to the revised version RPDEVS that incorporates a direct Mealy structure [3]. But as has been argued in [4], even then problems remain with the modeling and simulation of causal chains of concurrent events.

Therefore, the NSA-DEVS formalism (Non-Standard Analysis DEVS) is introduced in [4] that solves such problems with the drastic provision of prohibiting causal concurrent event chains altogether. To this end it introduces delay times at all inputs and forbids transitory states, i. e. states with a lifetime of zero. This allows to retain the mealy-type behaviour of RPDEVS, but makes the definition of an abstract simulator [5] much simpler than the corresponding simulator of RPDEVS [6].

Formally NSA-DEVS leads to the introduction of a large number of parameters for the necessary delays, whose usually very small values generally are of no interest at all. Therefore NSA-DEVS uses infinitesimal delays τ – often mostly given by a standard value τ_{def} –, and its abstract simulator clearly differentiates between the infinitesimal and finite time behaviour. This is possible in a mathematically precise way by resorting to the set ${}^*\mathbb{R}$ of hyperreal numbers, which are a well defined totally ordered field and form the basis of non-standard analysis [7]. ${}^*\mathbb{R}$ is an extension of the real numbers including the formal infinitesimal $\varepsilon > 0$, which is smaller than any positive real number, and the infinite number $\omega := 1/\varepsilon$. For the definition of NSA-DEVS one mainly needs the subset of positive finite hyperreals ${}^*\mathbb{R}^{>0}_{\text{fin}}$, occasionally enlarged by the single value ω, used as the lifetime of passive states.

The aim of this study is to further examine the practical usefulness of NSA-DEVS by implementing a set of examples with interesting event-cascades: The comparator-switch model from [8], a digital circuit containing flip-flops, a basic queue-server system and a complex queuing system. We will always start with a model that only contains default values for all input delays and transitory states, and then make the necessary fine-tuning to get the desired behaviour.This will show, whether the possible multitude of delay parameters can be tamed.

Another focus will lie on the exact definition of a model and its operation using the defined abstract simulator.

This will add confidence in the validity of the simulator algorithm, and show that modeling with NSA-DEVS – as with DEVS formalisms in general – can lead to a thorough understanding of a model and its behaviour.

1 The NSA-DEVS Formalism

The NSA-DEVS formalism is a variation of the basic PDEVS specification [1], which is divided into a model description and the definition of an abstract simulator. Two types of models are defined in PDEVS: an atomic model that describes the behaviour of a single component, and a coupled model, which shows how atomic models can be combined to build a hierarchical structure. The abstract simulator specifies the execution of a PDEVS model. It consists of three kinds of modules – a root coordinator, a coordinator for each coupled model and a simulator for each atomic model –, which exchange different types of messages to coordinate the behaviour of the atomic and coupled models.

In NSA-DEVS the definition of an atomic model is similar to the RPDEVS description, formally it is given by a 7-tuple $< X, S, Y, \tau, ta, \delta, \lambda >$ with

X	set of input ports and values,
S	set of states,
Y	set of output ports and values,
$\tau \in {}^*\mathbb{R}^{>0}_{\text{fin}}$	input delay time,
$ta : S \to {}^*\mathbb{R}^{\geq 0}_{\text{fin}} \cup \{\omega\}$	time advance function,
$\delta : Q \times X^+ \to S$	transition function,
$\lambda : Q \times X^+ \to Y^+$	output function.

For the definition of X and Y one uses sets P_{in} and P_{out} of input and output names and corresponding sets X_p and Y_p of possible values at input or output port p. Then the input and output sets are given as

$$\begin{aligned} X &= \{(p,v) | p \in P_{in}, v \in X_p\} \\ Y &= \{(p,v) | p \in P_{out}, v \in Y_p\} \end{aligned}$$

To describe the simultaneous arrival of input values at different ports, one additionally needs the set

$$X^+ := \big\{\{(p_1,v_1),\ldots,(p_n,v_n)\} \,|\, n \in \mathbb{N}_0,\, p_i \in P_{in},\, p_i \neq p_j \text{ for } i \neq j,\, v_i \in X_{p_i}\big\}$$

and similarly Y^+ for simultaneous outputs at several ports.

To specify the transition function δ and output function λ, which describe the basic behaviour of the model, one defines the set $Q = \{(s,e) | s \in S, 0 \leq e < ta(s)\}$ that combines a state and the elapsed time e since the last transition. As in RPDEVS, both event types (incoming event or internal state change) lead to a call of λ followed by a change to a new state according to δ.

The formal difference to RPDEVS is small, but important: All time values and intervals are meant here as subsets of the hyperreals ${}^*\mathbb{R}$, and the time advance function ta may be infinitesimal, but it is always > 0. A new element is τ, the delay time between the arrival of a set of inputs and the call of λ and δ. Generally it is an infinitesimal, often given by a default value $\tau_{def} = \varepsilon$, and is adapted if the need occurs.

A coupled NSA-DEVS model is defined as in PDEVS and RPDEVS, it consists of input and output ports and a set of atomic or coupled models, which are connected among themselves and to the external ports. The abstract NSA-DEVS simulator uses the same structure of modules and messages as in PDEVS, but it implements the input delays and the Mealy-like behaviour. Its details, the differences to the PDEVS simulator and an implementation in Matlab can be found in [5].

2 A Comparator-Switch Model

2.1 Theoretical analysis

In [8] a simple example is presented that consists of a switch controlled by a comparator (cf. Figure 1), such that an incoming entity is routed to the upper output, if it has a negative value, and to the lower output, if its value is positive or zero. As is shown in [8] this model can not be implemented straightforwardly in PDEVS using independent reusable components, but works perfectly well in RPDEVS. Therefore it is an excellent first example to test the capabilities of NSA-DEVS.

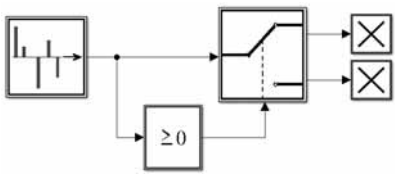


Figure 1: Example 1: Switch controlled by a comparator.

Using the NSA-DEVS specification of an atomic model, the comparator is defined by

$$P_{in} = \{in\},\ X_{in} = \mathbb{R} \Rightarrow X = \{(in,x) | x \in \mathbb{R}\}$$
$$P_{out} = \{out\},\ Y_{out} = \{0,1\} \Rightarrow Y = \{(out,0),(out,1)\}$$
$$S = \varnothing,\ ta(s) = \omega$$
$$\tau = \varepsilon \equiv \tau_{def}$$
$$\delta(s,e,x^+) = s$$
$$\lambda(s,e,\{(in,x)\}) = \begin{cases} \{(out,0)\} & | x < 0 \\ \{(out,1)\} & | x \geq 0 \end{cases}$$

The switch component needs an internal state to store the current routing behaviour of the switch, its formal definition is:

$$P_{in} = \{in, sw\},\ X_{in} = \mathbb{R},\ X_{sw} = \{0,1\}$$
$$\Rightarrow X = \{(in,x) | x \in \mathbb{R}\} \cup \{(sw,0),(sw,1)\}$$
$$P_{out} = \{out_1, out_2\},\ Y_{out1} = Y_{out2} = \mathbb{R} \cup \{\varnothing\}$$
$$\Rightarrow Y = P_{out} \times (\mathbb{R} \cup \{\varnothing\})$$
$$S = \{1,2\},\ ta(s) = \omega$$
$$\tau = r\varepsilon \quad (r = 1, \text{can be changed as parameter})$$
$$\delta(s,e,x^+) = \begin{cases} i+1 & | (sw,i) \in x^+ \\ s & | \text{otherwise} \end{cases}$$
$$\lambda(s,e,x^+) = \begin{cases} \{(out_s,x),(out_{3-s},\varnothing)\} & \\ \qquad\qquad | x^+ = \{(in,x)\} & \\ \{(out_{i+1},x),(out_{2-i},\varnothing)\} & \\ \qquad\qquad | x^+ = \{(in,x),(sw,i)\} & \\ \varnothing \qquad\qquad | x^+ = \{(sw,i)\} & \end{cases}$$

An important point to note is that the λ-function sends an empty value to the output port that is currently not used.

The example model contains two additional atomic components: a generator that outputs predefined values at given times, and a terminator that stores the last incoming value. Their formal definitions are straightforward and will not be needed in the folllowing. The complete model can then be easily defined as a coupled model.

To analyse the behaviour of this model in detail, we will retrace the simulator procedures. Generally, when an input value arrives at time $t = 1$ at a Mealy-like component with a time delay $\tau \in {}^*\mathbb{R}$, its simulator module S and the coordinator C of its enclosing coupled component perform the following steps [5]:

t	
1	C sends an x-message with the input value to S, which stores it in an internal variable x^* and sends back the time $1+\tau$ of its next internal event.
$1+\tau$	C sends a *-message to S, which now computes its output value $\lambda(s,e,x^*)$ and sends it to C via a y-message. C distributes it via x-messages to all connected components. After that C sends an empty x-message to S, which changes its state using $\delta(s,e,x^*)$ and resets x^* afterwards.

In the context of the example model the behaviour is more complicated. We will concentrate only on the switch with simulator S and assume that it is in state $s = 1$ (out_1 is active). At $t = 1$ the generator outputs a value $x = 1$. Now the following happens (cf. Figure 2):

t	
1	S gets $(in,1)$ via an x-message, stores it in x^* and returns $1+\varepsilon$.
$1+\varepsilon$	S gets a *-message, computes output $\{(out_1,1),(out_2,\varnothing)\}$ and sends it to C. Then the delayed output of the comparator arrives, therefore S doesn't get an empty x-message, but the input value $(sw,1)$. x^* now contains both values, the next event will be at $1+2\varepsilon$.
$1+2\varepsilon$	S gets a *-message, computes output $\{(out_1,\varnothing),(out_2,1)\}$ and sends it to C. Finally S gets an empty x-message, changes its state to $s = 2$ and clears x^*.

This chain of events boils down to the following behaviour: If a new input arrives during the waiting time of a component, it will either complement the output or overwrite it, and another delay time is added – which again could be extended. Consequently, intermediate outputs appear, only the final one representing the expected results. This behaviour reminds of the λ-iterations appearing in the RPDEVS simulator [6], but the underlying mechanism is quite different, as we will see in later examples.

Concentrating on the states the picture is simpler: The state only changes at the end of such an event chain and shows the anticipated behaviour. This becomes particularly clear, if one examines the terminator blocks: Though the value 1 appears at both outputs of the switch, it is only stored in the state of the (cor-

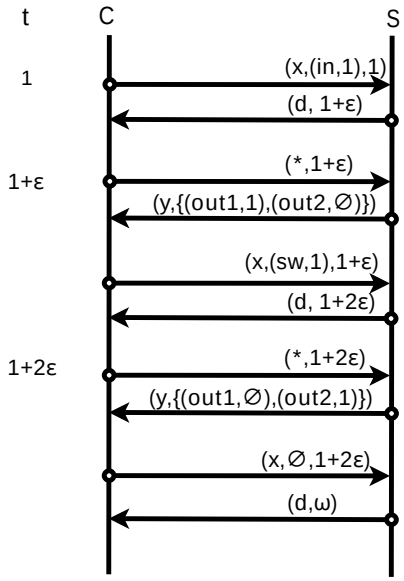


Figure 2: Internal messages from or to the Switch simulator.

rect) lower terminator. The incoming value in the upper terminator is erased by the empty second output and doesn't reach the state variable. To make this happen, it is important that the λ-function of an atomic component explicitely outputs empty values at unused ports.

2.2 Practical implications

What this complex behaviour means for a practitioner, who is not interested in the internal workings of the simulator, depends on the output values that a concrete simulator environment provides. In the PDEVS simulator hyPDEVS [9], a simulation run produces output values representing the states of the atomic components. In this case a similar NSA-DEVS simulator would show exactly the expected behaviour of the comparator-switch model.

The NSA-DEVS simulator that is described in [5] has no intrinsic output possibilities. Instead it uses an atomic component *ToWorkspace* that can be connected to an output port and copies the incoming values to a global output variable, which can be plotted or analysed after the simulation run. Like all NSA-DEVS atomics it has an internal delay τ. If one sets $\tau = r_{Out}\varepsilon$ for all ToWorkspace blocks and chooses a value r_{Out} that is larger than the delays of all other atomics, the intermediate outputs do not show up in the global output variables and the behaviour is again as expected. Choosing a very small value for r_{Out}, one can make these outputs visible, which could be useful for debugging purposes.

Another possible approach would be to set the delay of the switch component to 2ε. Since the input from the comparator now arrives, before the *-message of the switch is called, no intermediate outputs are generated and one can use arbitrary (especially: default) delays for ToWorkspace blocks. Thinking along these lines, one could take the appearance of intermediate outputs as a hint to properly adapt some delays.

3 Flip-flops and Shift Register

As has been shown in [10], the modeling of simple digital circuits containing flip-flops can be a challenge for discrete-event based systems. A solution for the case of RPDEVS has been given in [11]. Using the example of a simple shiftregister (cf. Figure 3), we will demonstrate in the following that NSA-DEVS can cope with such examples easily.

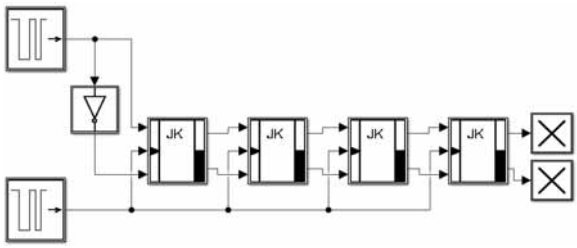


Figure 3: Example 2: Shiftregister with four JK flip-flops.

The basic component is the JK flip-flop, which has three binary inputs (J, CLK, K), two outputs $(Q, \overline{Q})$ and four internal states, three of them to store incoming values, and one for the proper state of the flip-flop. When the *CLK* input switches from 1 to 0 (a "clock tick"), the state changes according to the following function:

$$f(j,k,q) = ((j \vee q) \wedge \bar{k}) \vee (j \wedge k \wedge \bar{q}),$$

where j,k are the input values and q is the previous value of the internal state. Very important is the correct behaviour, when the arrival of inputs coincides with a clock tick: In this case the old (stored) values are used to compute the next state, after that the new values are stored internally. This is necessary to implement the correct behaviour of a shift register, where incoming values are shifted at a clock tick for one step along the line of flip-flops.

These considerations lead to the following formal definition of the JK flip-flop atomic model:

$$\mathbb{B} := \{0,1\}$$
$$P_{in} = \{J, CLK, K\},\ X = P_{in} \times \mathbb{B}$$
$$P_{out} = \{Q, \overline{Q}\},\ Y = P_{out} \times \mathbb{B}$$
$$S = \mathbb{B} \times \mathbb{B} \times \mathbb{B} \times \mathbb{B},\ \text{where } s \equiv (j, clk, k, q) \in S$$
$$ta(s) = \omega$$
$$\tau = r\varepsilon \quad (r \text{ parameter})$$
$$\delta(s, e, x^+) = (j', clk', k', q') \text{ with}$$
$$q' = \begin{cases} f(j,k,q) & | clk = 1 \wedge (CLK, 0) \in x^+ \\ q & | \text{otherwise} \end{cases}$$
$$j' = \begin{cases} b & | (J, b) \in x^+ \\ j & | \text{otherwise} \end{cases}$$
$$clk' = \begin{cases} b & | (CLK, b) \in x^+ \\ clk & | \text{otherwise} \end{cases}$$
$$k' = \begin{cases} b & | (K, b) \in x^+ \\ k & | \text{otherwise} \end{cases}$$
$$\lambda(s, e, x^+) = \begin{cases} \{(Q, f(j,k,q)), (\overline{Q}, \overline{f(j,k,q)})\} & \\ & | clk = 1 \wedge (CLK, 0) \in x^+ \\ \varnothing & | \text{otherwise} \end{cases}$$

The complete example model is a coupled model that contains two binary generators producing test inputs, a not gate, four JK flip-flops and two terminators. The formal description of all these models is straighforward. Using the Matlab-based NSA-DEVS simulator from [5] the complete model is easily implemented and run.

Adopting the global default value for all input delays, the simulation results are as expected, no twisting of any parameters is necessary. This example shows that the key to a valid implementation of flip-flops is a precise definition of their behaviour – and that the NSA-DEVS formalism offers the tools do this easily.

4 A Simple Queue-Server System

The third example consists of a generator that creates entities in fixed time intervals $t_G = 1$ and sends them to a queue, which is connected to a simple server with fixed service time t_S = 1.5. Entities leaving the server are terminated (cf. Figure 4).

The queue outputs entities unless it is blocked; its blocking status is given by an additional input coming from the server. This model has already been used as a case study in [4] and [5], it shows a complex behaviour due to its cascade of causally related concurrent events. The queue and server components have an additional output for the number of stored entities (y_{nq} or y_{ns}, resp.), which will be used in the last example model.

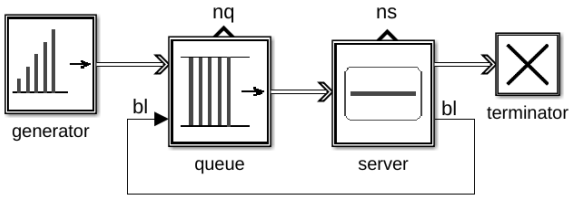


Figure 4: Example 3: Singleserver model combining a queue and a server component

The formal mathematical definition of the components is straightforward, but a bit cumbersome, especially for δ and λ. Therefore, their behaviour will be described by an enhanced state diagram like in Figure 5 for the server. These "macroscopic" states are not identical to the NSA-DEVS states, i. e. elements of the set S, which usually is much larger, but contain the essential information to conveniently describe the behaviour of the component.

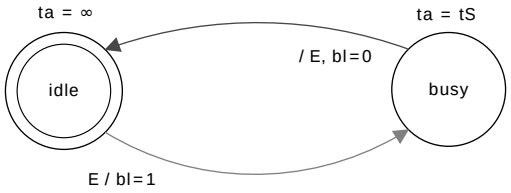


Figure 5: State diagram of the server component.

Nevertheless, it is a useful exercise to explicitely write down most of the formal structure of a component, to make its definition as precise as possible. For the server component this could be done in the following way (identifying an entity with a real number for simplicity):

$$P_{in} = \{in\},\ X_{in} = \mathbb{R} \Rightarrow X = \{in\} \times \mathbb{R}$$
$$P_{out} = \{out, ns, bl\}, Y_{out} = \mathbb{R}, Y_{ns} = \{0,1\} = Y_{bl}$$
$$\Rightarrow Y = \{out\} \times \mathbb{R} \cup \{\ (ns,0),(ns,1),(bl,0),(bl,1)\})$$
$$S = \{(\varnothing, \omega)\} \cup (\mathbb{R} \times [0, t_S])$$
$$ta((v,\sigma)) = \sigma$$
$$\tau = r\varepsilon \quad (r \text{ parameter})$$

While most of this definition is straightforward, the set of states needs some explanations: The state consists of the value v of a stored entity (or $\varnothing$) and the current lifetime σ of the state. This is a frequently used trick and reduces the time advance function to simply returning σ. It is necessary here to cope with a special situation: When an entity reaches the input, while the server is busy, the entity is discarded, and the waiting time of the currently stored entity has to be reduced. This can easily be done by changing the value of σ. Note that the "macroscopic" states *idle* and *busy* are only implicitely given by

$$idle \equiv (v = \varnothing)$$
$$busy \equiv (v \in \mathbb{R})$$

A crucial point in the definition of δ and λ is to take all possible values of $Q \times X^+$ into account. A helpful approach here is to divide these values into internal, external and confluent events – just as in PDEVS –, using the values of the function arguments s, e and x^+.

The definition of the queue can be done along these lines using the state diagram in Figure 6. Four macroscopic states are defined according to the blocking status and the size of the queue (empty or not). The critical state here is *queuing free*, which is the only state, where the queue outputs entities. It is a transitory state, which in the context of NSA-DEVS becomes a state with an infinitesimal delay $r_d\,\varepsilon$. The value of r_d is defined as a parameter with the usual default value of 1; as we will see, it plays a crucial role in the correct implementation of the singleserver example.

If one runs the complete model on the Matlab-based NSA-DEVS simulator, using default parameter values for all input delays and r_d, the result is incorrect: At time $t = 4$ the queue sends the fourth entity to the server instead of the third, which gets lost (cf. Figure 7). The basic reason for this behaviour is evident: After sending entity three, the queue stays in the state *queuing free* and sends the next entity after a delay of ε, because the

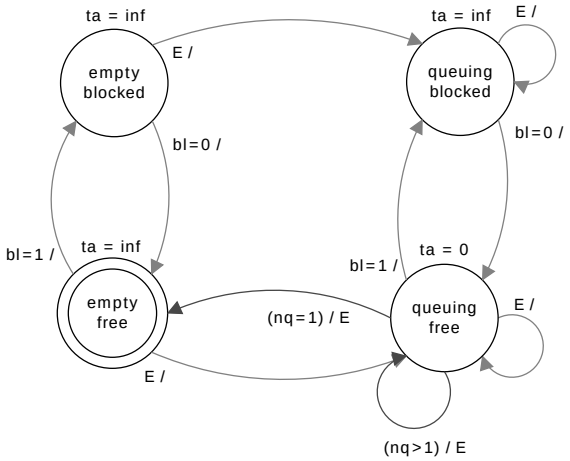


Figure 6: State diagram of the queue component.

blocking signal from the server has not arrived yet. In order to guarantee the desired ordering of events, one simply has to properly enlarge the value of r_d to get the expected simulation results.

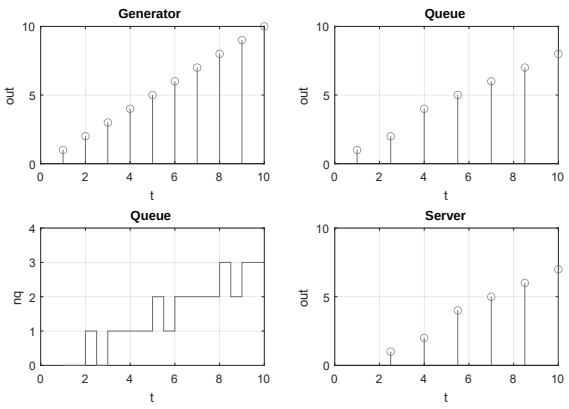


Figure 7: Simulation results of singleserver with default parameters.

Simple considerations like these are usually sufficient to cope with most problems coming from a wrong ordering of concurrent events. But if one looks more closely, there are still some fine points that are not immediately obvious, such as: Why are the results shown in Figure 7 so much different from the corresponding results in [4, Fig. 6]? And why is a value of $r_d = 1.1$ already large enough to produce correct results? One can always use the debugging features of the simulator to retrace its behaviour, which makes answering such questions a straightforward, if tedious, exercise [12].

5 A Complex Queuing System

The final example is a simplified version of the basic queuing system from the Argesim benchmark C22 [13]. It contains a generator and a set of three queue-server lines. Incoming entities are routed to the shortest line (including the server allocation) and leave the system after being served (cf. Figure 8). The benchmark defines two model variations according to the order of concurrent events: In variant A an entity leaves a server, before a new entity enters the system, in variant B the order is reversed.

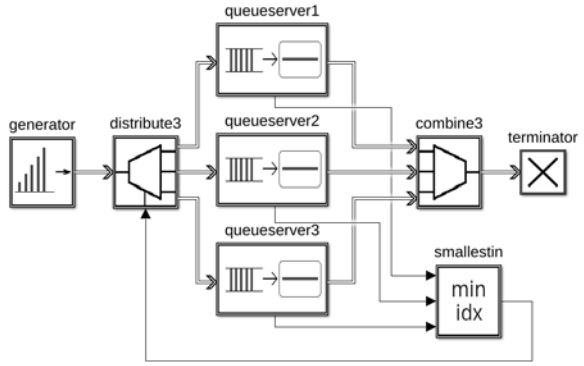


Figure 8: Example 4: Queuing system fifo3 containing three queue-server lines.

The queue-server lines are defined as coupled models consisting of the queue and server components from above and a simple atomic block that adds their loads. An additional ToWorkspace block is added for logging purposes (cf. Figure 9). According to the lesson learned from the last example, all input delays are identical, while the delay time of the transient queue state is twice as large.

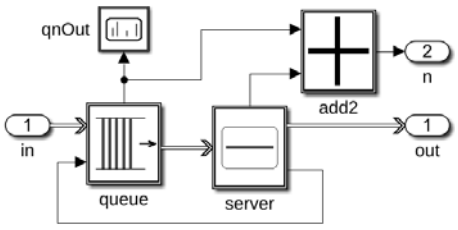


Figure 9: Internal structure of coupled model queueserver.

Three more atomic components are needed to complete the example:

- a distributor that routes incoming entities to the output that is defined by a control input,
- a combiner that accepts entities from its inputs and sends them to its single output,
- a computational block (smallestin) that gets the three loads, computes the minimal value and outputs the number of the smallest input where the minimum occurs.

They all can be implemented easily, the only interesting one is the combiner: When entities appear concurrently, they will be stored internally and output one after the other. The necessary transitory state as always induces an (infinitesimal) delay between the outgoing entities. If one uses a ToWorkspace block to display them, one has to set its delay to a small value, since otherwise one would only see the last outgoing entity.

Using only standard parameters for all delays, the complete model works without further ado and produces the expected results (cf. Figure 10). Having a close look at the order of the outgoing entities, one finds that the model realizes the variant B: Due to the delays of the adder and the smallestin block, the information that a server is empty arrives at the distributor after the new entity from the generator has passed. To implement variant A, one simply increases the delay of the distributor either to an arbitrary large value or – after chasing the delays through the diagram – to 3ε.

6 Conclusion

As the careful analysis of the examples has shown, the expected large number of delay parameters needed in NSA-DEVS usually boils down to one default value and a bit of fine-tuning in special cases. The most notable exception was the queue, where the lifetime of the transitory state has to be enlarged. But since a queue component with a properly adapted default value will usually be part of the model library, a user in practice won't come in touch with this exception.

Two simple provisions have been identified that often will help to hide the internal details: Firstly, if a component with several output ports sends values only to some of them, it should send an empty value to the remaining ports, in order to get rid of intermediate output values. In a Matlab implementation this could be the empty array `[]`. Secondly, one should choose a large input delay for the ToWorkspace blocks, which again could be predefined already in the model library.

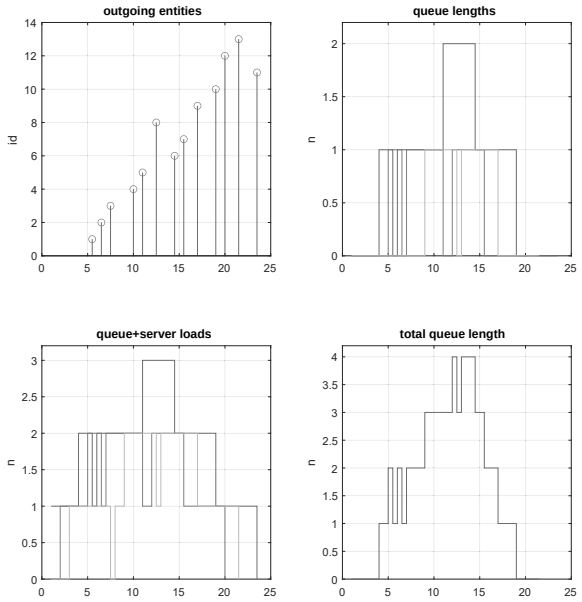


Figure 10: Simulation results of fifo3 with default parameters.

Of course there are situations, where one needs to think about the infinitesimal behaviour, e. g. to define a special ordering of concurrent events. Another example occurs in the comparator-switch model, when a chain of components before the comparator input leads to a large total delay, which has to be compensated by the delay of the switch. But such difficulties are basically inherent to discrete-event modeling, and the question is not, how to avoid them, but how to cope with them in a clear-cut way. One could argue that NSA-DEVS provides the right balance between hiding details in simple cases and giving access to the internals to solve these problems.

An open question at the end of [5] was, whether one delay time for an atomic model suffices, or if one needs port specific delay times. So far, the simple definition used in NSA-DEVS seems to work generally. If necessary, a simple workaround would be, to add a special delay component – basically a gain with factor 1 – before a port that needs a special delay. This question could be further examined in the context of the final remaining task from the todo list at the end of [4], which was the implementation of a complex case study in NSA-DEVS, and will be adressed in a future investigation.

References

[1] Zeigler BP, Muzy A, Kofman E. *Theory of Modeling and Simulation*. San Diego: Academic Press, 3rd ed. 2019.

[2] Goldstein R, Breslav S, Khan A. Informal DEVS conventions motivated by practical considerations. In: *Proc. of Symposium on Theory of Modeling &Simulation – DEVS Integrative M&S Symposium*. 2013; pp. 10:1–10:6.

[3] Preyser FJ, Heinzl B, Kastner W. RPDEVS: Revising the Parallel Discrete Event System Specification. In: *9th Vienna Int. Conf. Mathematical Modelling*. Wien. 2018; pp. 242–247.

[4] Junglas P. NSA-DEVS: Combining Mealy Behaviour and Causality. *SNE Simulation Notes Europe*. 2021; 31(2):73–80. doi: 10.11128/sne.31.tn.10564.

[5] Jammer D, Junglas P, Pawletta T, Pawletta S. A Simulator for NSA-DEVS in Matlab. In: *Proc. of ASIM 2022 – 26. Symposium Simulationstechnik*. Wien. 2022; pp. 93–100. doi: 10.11128/arep.20.a2005.

[6] Preyser FJ, Heinzl B, Kastner W. RPDEVS Abstract Simulator. *SNE Simulation Notes Europe*. 2019; 29(2):79–84. doi: 10.11128/sne.29.tn.10473.

[7] Goldblatt R. *Lectures on the Hyperreals*. New York: Springer. 1998.

[8] Preyser FJ, Heinzl B, Raich P, Kastner W. Towards Extending the Parallel-DEVS Formalism to Improve Component Modularity. In: *Proc. of ASIM-Workshop STS/GMMS*. Lippstadt. 2016; pp. 83–89.

[9] Pawletta T, Deatcu C, Pawletta S, Hagendorf O, Colquhoun G. DEVS-based modeling and simulation in scientific and technical computing environments. In: *Proc. of DEVS Integrative M&S Symposium (DEVS'06) - Part of the 2006 Spring Simulation Multiconference (SpringSim'06)*. Huntsville/AL, USA: D. Hamilton. 2006; pp. 151–158.

[10] Junglas P. Pitfalls using discrete event blocks in Simulink and Modelica. In: *Proc. of ASIM-Workshop STS/GMMS*. Lippstadt. 2016; pp. 90–97.

[11] Fiedler C, Preyser FJ, Kastner W. Simulation of RPDEVS Models of Logic Gates. *SNE Simulation Notes Europe*. 2019;29(2):85–91. doi: 10.11128/sne.29.tn.10474.

[12] CEA Wismar. *NSA-DEVS on GitHub*. `https://github.com/cea-wismar/NSA-DEVSforMATLAB`.

[13] Junglas P, Pawletta T. Non-standard Queuing Policies: Definition of ARGESIM Benchmark C22. *SNE Simulation Notes Europe*. 2019;29(3):111–115. doi: 10.11128/sne.29.bn22.10481.

Reviewing Recommender Systems in the Medical Domain

Katharina Brunner[1,2], Bernhard Hametner[2*]

[1] Institute of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8-10

[2] AIT Austrian Institute of Technology, Center for Health and Bioresources, Medical Signal Analysis, Vienna, Austria.
[*] *bernhard.hametner@ait.ac.at*

SNE 32(4), 2022, 203-209, DOI: 10.11128/sne.32.tn.10624
Received: 2022-04-08; Revised:2022-06-26
Accepted: 2022-07-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Medical recommender systems are increasing in popularity within the digital health sector. Two main principles for personalised support are just-in-time interventions, and adaptiveness of treatment. Intervention concepts using these principals are called JITAIs, and they aid clients in self-management for health-related issues. In this contribution, the JITAI framework is introduced, and its advantages for recommender systems are discussed. Mathematically, the JITAI concept can be interpreted as a contextual or regular multi-armed bandit problem, which is solved via a bandit algorithm. After discussing several algorithmic strategies of bandit algorithms and elaborating on their differences, the Thompson Sampling strategy is identified as a practical solution for real-life applications using the JTIAI framework. Subsequently, existing recommender systems based on the (contextual) multi-armed bandit approach are reviewed, and the disruption of the algorithm's learning process by instances of missing data is found to be a prevalent obstacle. An algorithm called Thompson Sampling with Restricted Context is put forward as a solution, where missing data is processed within the bandit setting.

Introduction

Digital health practices are expected to revolutionise the public health sector on a global scale and provide healthcare solutions to all people regardless of geographic location and social strata. In digital health, data is processed with the help of smart and connective devices using advanced computing and artificial intelligence, including machine learning as well as other data science strategies [1].

A research domain within this framework is the development of personalised recommender systems, with the aim to support the self-management of chronic illnesses, or the facilitation of building habits towards a more healthy lifestyle. Generally, two principles are combined: just-in-time assistance, and adaptiveness. Recommender systems based on this approach are called just-in-time adaptive interventions (JITAIs) [2]. Some JITAIs employ machine learning techniques to identify the best supportive intervention for the client out of a pool of possible options, and deliver it at a time when the client is most receptive, or has the highest need for it. Mathematically, this problem can be interpreted as a contextual multi-armed bandit problem, which has its origins in game theory, and aims to find the option that yields the highest reward under given circumstances [3].

This contribution introduces the most common approaches (i.e., algorithms) to solve the problem and provides an overview of the state-of-the-art mobile health applications that operate using the (contextual) multi-armed bandit approach.

1 The JITAI Framework

A pragmatic framework is provided by Nahum-Shani et al. [6], which can help developers in constructing JITAI intervention concepts, or may inform the design of a JITAI model. JITAIs are multi-component interventions, and consist of five key elements: *decision points*, *tailoring variables*, *intervention options*, *decision rules*, and *proximal outcomes*. At a decision point (i.e., a point in time at which an intervention decision is made), the decision rules determine which intervention option out of an array of possible candidates is best to facilitate a proximal outcome (i.e., the short-term goal the intervention is trying to achieve), based on the contextual data concerning the client, which is stored in tailoring variables.

Together, these components form an intervention concept, see Figure 1.

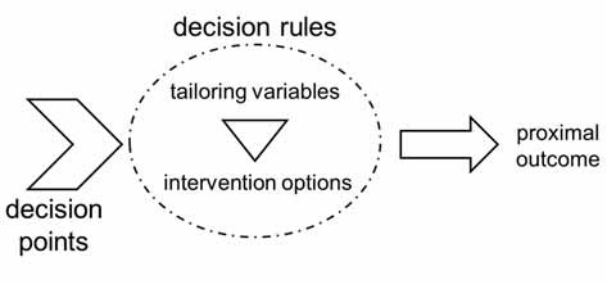


Figure 1: Intervention concept for the JITAI design, adapted from [2].

This definition of the JITAI framework allows for a variety of applicative options within each component. For example, decision points may occur at pre-specified time intervals (e.g., every three minutes), at specific times of day (e.g., daily at 9 a.m.), or following random prompts, depending on how frequent meaningful changes in the tailoring variables are expected to take place.

In mathematics and many mathematics-adjacent fields of application, the JITAI framework can be constructed as a contextual or regular multi-armed bandit problem (see below), due to both concepts being translatable in a natural way.

2 The Multi-Armed Bandit Problem

The multi-armed bandit (MAB) problem is perhaps the simplest model for sequential decision making where the aim is to maximise the cumulative sum of rewards over a certain time horizon [4].

The original setup describing the MAB problem shows a player who is faced with k slot machines (colloquially known as one-armed bandits) [5]. Identically, one can imagine a single slot machine with several arms, thus obtaining the term *multi-armed bandit*. The player aims to maximise the cumulative reward from playing the machines over a certain number of plays, which holds the intrinsic dilemma of the MAB problem: the exploration-exploitation trade-off. The player needs to balance between trying out all arms sufficiently often to discover which ones are most lucrative (exploration), while concurrently playing the arms that they have found to yield the highest rewards (exploitation).

General approaches of the MAB problem call the playing entity the *agent*, and the process of systematically playing towards a specific goal is done by a MAB algorithm. The contextual MAB (CMAB) setting is an extension of the MAB problem where the player views additional information about the current situation before deciding which arm to play, thereby avoiding unnecessary exploration, and is guided towards the arms that need to be explored. Therefore, CMABs lend themselves to applications in the medical setting, where any recommender system should base their intervention decisions on the observed health data. In general, CMAB algorithms are derived from MAB algorithms, so the algorithmic strategies of the MAB problem are a good entry point into applied bandit algorithms.

2.1 Bandit Algorithms

A possible option for MABs is to consider the stochastic bandit setting, where the agent chooses an arm at an iteration point t and subsequently receives a reward drawn from an arm-specific distribution unknown to the agent. The agent then improves its arm selection strategy based on the observation, with the goal to estimate parameters that describe the distributions linked to the arms, see Algorithm 1 [7]. Then, the agent will exploit the arm that is estimated to yield the highest reward. Usually, fixed distributions are assumed for all arms, and the performance of the algorithm can be quantified by observing how quickly the optimal arm is identified through the obtained reward, or, alternatively, the concept of regret [4].

Algorithm 1: Multi-Armed Bandit Algorithm

Input: $\mathscr{A} = \{A_1, \ldots, A_k\}$, $\mathscr{R}_1, \ldots, \mathscr{R}_k$
for $t = 1, 2, \ldots$ **do**
  choose arm $a_t \in \mathscr{A}$
  receive reward $r_t \sim \mathscr{R}_{i(t)}$
  improve arm-selection strategy with new observation (a_t, r_t)
end

Since the true reward distributions are unknown to the agent, MAB algorithms estimate the expected reward [8], and their overall performance can be evaluated by regarding the cumulative sum of rewards over all iterations. Instead of observing how well the algorithm has done, one can also investigate how often the algorithm has missed out on the optimal arm.

At iteration t, the optimal arm is defined as the arm with the (currently) highest expected reward estimate, and the regret at t is the difference between the optimal choice and the arm that the MAB algorithm has actually chosen.

As an example for a bandit setting, consider the Bernoulli bandit problem (a special case of the MAB problem), where all k reward distributions are Bernoulli distributions. The reward $r_{i(t)}$ the agent receives when choosing arm $a_t \in \mathscr{A}$ at time t is:

$$r_{i(t)} = \begin{cases} 1 & \text{with probability } p_i \\ 0 & \text{with probability } 1-p_i \end{cases}, \quad p_i \in (0,1)$$

The parameters p_i, $i \in \{1,\ldots,k\}$ are unknown to the agent [9]. In case of a Bernoulli bandit, the expected reward for chosen action a_t at t is the success probability for the arm:

$$\mathbb{E}\left[r_{i(t)}\right] = p_{a_t}$$

After choosing an arm, the agent either succeeds ($r_{i(t)} = 1$) with probability p_i, or fails ($r_{i(t)} = 0$) with probability $(1-p_i)$. Since the expected reward of the optimal arm is $r_{a_{opt}} = \max_i(p_i)$, the regret $R_{a(t)}$ of choosing a suboptimal arm at time t is denoted by:

$$R_{a(t)} = \max_i(p_i) - p_{a(t)}$$

Different simulation runs of the Bernoulli bandit can be rated in their efficiency: the lower the cumulative regret, the more efficient the algorithm is in finding and exploiting optimal options, thus exemplifying how regret can be practical when comparing different algorithmic strategies. Even though the Bernoulli bandit gives the most basic bandit setting, it can be found in many practical applications, and more complex MAB problems may be simplified by assuming a Bernoulli distribution instead of a more advanced one [8].

2.2 Algorithmic Strategies

The main obstacles for solving the Bernoulli bandit problem are the unknown success probabilities, because the expected regret cannot be computed. However, when determining algorithmic performance in a simulation environment, the success probabilities are known, even though they are assumed not to be, in order to assess the quality of different algorithmic strategies. For a good strategy, the regret rapidly decreases to zero, thus the best option is identified quickly, and subsequently exploited. In contrast, if the value of regret decreases slowly or never reaches zero, the strategy is considered poor. Figure 2 shows a sketch of possible regret curves.

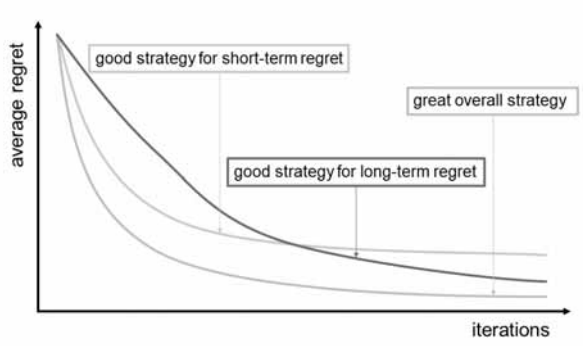


Figure 2: Possible regret curves for different MAB algorithm strategies.

Different strategies are distinguished by how the exploration-exploitation trade-off is handled. Three potential strategies are: the *e-Greedy strategy*, the *Upper Confidence Bound* (UCB) *strategy*, and the *Thompson Sampling* (TS) *strategy*.

The e-Greedy Strategy. The e-Greedy strategy explicitly trades off between exploration and exploitation, by using the exploration parameter e. A greedy strategy refers to exploitation without exploring, i.e., choosing the arm with the highest current reward estimate. This bears the risk of missing the optimal arm forever, see the following example:

Let $\mathscr{A} = \{A_1, A_2, A_3\}$ be a Bernoulli bandit with three arms, and let the (true) success probabilities be:

$$p_1 = 0.3, \quad p_2 = 0.7, \quad p_3 = 0.8$$

Furthermore, let the initial success estimates be equal for all three arms:

$$e_1^0 = 0.5, \quad e_2^0 = 0.5, \quad e_3^0 = 0.5$$

Here, arm A_1 is overestimated, whereas arms A_2 and A_3 are underestimated. Since all arms have equally high success estimates, the agent picks one arm at random. Let A_2 be the agent's choice, and let $r_1 = 1$. The estimates are updated to:

$$e_1^1 = 0.5, \quad e_2^1 = 0.75, \quad e_3^1 = 0.5$$

Following a greedy strategy, the agent picks A_2 again, and A_3 (the optimal arm) will only be explored if the estimate for A_2 drops down to 0.5, which is not likely to happen, due to the true success probability being 0.7.

This problem is solved by introducing an exploration parameter $e \in (0,1)$, which sets the probability of performing an exploration step at t, wherein one arm is chosen randomly. However, the risk remains that the exploration continues after having identified the optimal arm, since the algorithm forces the agent to select a (known) suboptimal arm during each exploration step, thus the regret will never converge to zero.

An alternative is presented by the decaying e-Greedy strategy, where e is not fixed, but decays over time. However, an accurate value for the decay is difficult to determine.

The Upper Confidence Bound Strategy. The UCB strategy deals with the exploration-exploitation trade-off in an implicit way. In the previous strategy, the agent's knowledge at time t is modelled as a point estimate, which does not reflect the uncertainty regarding this value. In contrast, the UCB strategy explicitly models the knowledge uncertainty as confidence intervals, where both the current knowledge (i.e., the mean) and the related uncertainty (i.e., the width of the confidence interval) are used to guide the arm selection process.

In case of the Bernoulli bandit example, the probability estimates e_1^t, e_2^t, e_3^t, are substituted by UCBs for each arm. This principle is called "optimism in front of uncertainty": the uncertainty about the expected reward is expressed as a confidence interval, and the expected reward is estimated optimistically as the upper bound of that confidence interval.

Thus, there are two reasons why the UCB is high: the arm has not yet been explored, resulting in much uncertainty about the success probability, or the arm has been found to be a good choice, thus there is little uncertainty about the (high) success probability. This way, the agent keeps exploring arms that have not yet been proven to yield low rewards instead of arms that produce low rewards with high certainty [10].

There are different ways to derive UCBs, for example via the Hoeffding equation, or the Bayes theorem. However, computing UCBs can be difficult, depending on the assumed distributions.

The Thompson Sampling Strategy. The TS strategy works similarly to the previous strategy, but an agent following this strategy picks an arm randomly, according to its probability to be the best. The Bayesian update rule, which is derived from the Bayes theorem, lies at the heart of the TS strategy. A general formulation of the Bayesian update rule is

$$\text{Posterior} \propto \text{Likelihood} \times \text{Prior}$$

In practice, it means that, once an arm is chosen, the estimates representing the distribution of the arm are updated with the help of the reward observation. The conjugate property of prior-likelihood combinations plays an important part when updating distributions. The use of the Bayesian update rule is only recommended if the updated distribution (i.e., the posterior) is easily calculated, which is the case for conjugate prior-likelihood combinations. In a Bernoulli bandit, the conjugate combination is given by the Beta distribution.

TS is best explained when investigating the Beta-Bernoulli prior-likelihood combination. The Bernoulli distribution depends on the success parameter p, which needs to be estimated. The Beta distribution as the prior (and posterior) represents the uncertainty about p. Its parameters α and β correspond to successful and failed draws: if $r_t = 1$, α is upped by 1, and if $r_t = 0$, β is upped by 1, according the Bayesian update rule. The density function of the Beta distribution changes with each reward observation whenever the arm is chosen, and congregates around the estimated success parameter $e^t \sim p$. Instead of calculating statistical quantities concerning p, it is sufficient to draw a random variable from the Beta distributions at t, and the arm with the highest sample value is chosen by the agent, see Algorithm 2.

Empirical evaluation has shown that TS algorithms are more robust against delayed, or batched, feedback in applications for advertising and news article recommendations modelled as a CMAB problem [11], and that it has lower regret in the long run compared to UCB algorithms [12]. Even though the theoretical understanding of TS is still limited, optimal regret bounds on the expected regret exist for the MAB problem with Bernoulli distributions [8], and theoretical guarantees are provided for a TS algorithm equipped to solve the CMAB problem [13].

Algorithm 2: Thompson Sampling strategy

Input: $\mathscr{A} = \{1,\dots,k\}$, initial parameters α_i, β_i, auxiliaries $S_i = F_i = 0$
for $t = 1,2,\dots$ **do**
  for $i = 1,\dots,k$ **do**
    Draw θ_i according to Beta$(\alpha_i + S_i, \beta_i + F_i)$
  end
  choose arm $a_t = j = \text{argmax}_i\, \theta_i$
  receive reward $r_t \sim$ Bernoulli(e_j)
  if $r_t = 1$ **then**
    $S_j = S_j + 1$
  else
    $F_j = F_j + 1$
  end
end

Furthermore, the applicability of the TS strategy to any conjugate prior-likelihood combination facilitates the increase of model complexity beyond Bernoulli distributions without increasing computational complexity [9].

3 JITAIs and CMABs

The CMAB problem setup provides a natural model for developing digital health interventions of the JITAI design. In the previous section, MAB algorithms are discussed as a way of solving the MAB problem. From this, three main elements in a MAB algorithm can be derived: points in time (also *trials*), a set of arms, and respective reward distributions. The contextual bandit setting adds one more element: a context vector $x_t \in \mathbb{R}^d$, which holds the additional information that the agent views before selecting an arm during each trial. Depending on the chosen method of implementation, exploration and exploitation are balanced in order to minimise the cumulative expected regret.

Section 1 introduces the five components of the JITAI design, and explains how they interact. There are similarities in the formulation of both concepts, making the translation easy: decision points denote the trials, tailoring variables represent any contextual information in the form of a vector, and the possible intervention options serve as the arms of a bandit. Reaching the proximal outcome is equal to minimising the regret at t, or, in case of a Bernoulli bandit, succeeding at a trial [14]. The decision rules can be viewed as a mapping between the current values of the tailoring variables and the intervention options. This mapping is done by the bandit algorithm. Table 1 summarises the analogies.

JITAI framework	CMAB concept
Decision points	Trials
Tailoring variables	Context vector
Intervention options	Arms
Proximal utcome	Rewards
Decision rules	Bandit algorithm

Table 1: Representation of the analogy between the elements of the CMAB approach and the components of the JITAI design.

Due to its setup and adaptability, the JITAI design is favoured for personalised recommender systems. Since CMABs are most convenient for algorithmic implementation of JITAIs, by transitivity, recommender systems are best realised via implementing and solving a CMAB problem.

4 Medical Applications using (C)MABs

It is already common practice to use (C)MAB algorithms for researching personalised adaptive interventions in digital health. In the past decade significant progress has been made in creating functional applications that work in a (C)MAB setting, adapting to a client's intervention preferences in real time, as part of the trend towards reinforcement learning methods.

Medical recommender systems are multi-faceted, and the algorithm for intervention decisions is only one cog in a delicate machine. Mechanisms must be in place so that the decision rules can still function in case of missing, or erroneous, data, without compromising the algorithm's learning process. Clients must be kept engaged beyond an initial novelty period, so behavioural psychology plays an important part in the delivery of intervention suggestions to ensure that intervention engagement prevails over intervention fatigue [2]. Recommender systems intended for client use address these issues in different ways.

HeartSteps. HeartSteps is a mobile phone application currently available for download in the United States. Originally tested during at trial for improving physical activities of individuals with blood pressure in the stage 1 hypertension range, it delivers activity suggestions to encourage walking while monitoring the client's daily step count with a Fitbit tracker.

The intervention decisions are made via a CMAB algorithm that uses TS and the application is designed to include the delayed effect of treatment. However, the chosen algorithm cannot deal with missing data within the decision rules. Instead, a lack of data is compensated outside of the bandit algorithm [15].

MyBehavior. MyBehavior is a mobile phone application that delivers personalised interventions for promoting physical activity and dietary health as a JITAI, via a MAB algorithm. Phone sensory data is used to design unique recommendations for a client, with the goal to find activity suggestions that maximise the chance of daily calorie burns. The application records data every minute, and issues an activity suggestion once each morning. It then analyses the location tagged activities to find patterns that are representative of the client's behaviour. Additionally, MyBehavior allows clients to self-report exercise and food intake, which is backed by a crowd-sourcing database. Like HeartSteps, the decision rules cannot compensate for instances of missing data [16].

PopTherapy. PopTherapy is a mobile phone application that helps clients cope with stress and depression-related symptoms based on cognitive behavioural theory technology. Intervention suggestions are issued after a request is prompted by the client, and the goal is to maximise stress reduction. The application uses a CMAB algorithm combined with the UCB approach to select an intervention from a series of stress management strategies. However, the bandit algorithm requires knowledge of the correct model for the reward function [17] and, as is intrinsic to the UCB strategy, bases its arm selection process deterministically on historical data [15].

5 Conclusion and Outlook

Digital health is currently at the forefront of biomedical research, with recommender systems promising easier access to treatment for a variety of chronic illnesses, mental health challenges, and general life improvements. Intervention concepts that combine just-in-time support and adaptiveness of treatment (JITAIs) are aiming to provide personalised support to clients at points in time when they need it most, or are most receptive to it, based on the processing of health data.

As can be seen from existing recommender systems, the mathematical concept of (C)MABs is a convenient way of implementing JITAIs as real-life applications.

Different strategies for (C)MAB algorithms are available, and a thorough comparison of the most common approaches shows that the TS strategy stands out in terms of arm selection performance, adaptivity, and computational effort. Theoretical knowledge about TS is still sparse. However, in setups where regret is bounded, the algorithm is guaranteed to identify arms that are close to optimal options eventually. Thus, when investigating algorithmic strategies for JITAI-backed recommender systems, consideration should be given to TS above others, when applicable to the research question.

Furthermore, consideration should also be given to an extension of TS called *Thompson Sampling with Restricted Context* by Bouneffouf et al. [18], where the agent only observes limited context (i.e., a restricted context vector) at the cost of a slight decrease in performance. This setting offers a natural way of dealing with instances of missing data by simply disregarding it, and the lack of data can be addressed within the bandit setting while the learning process of the bandit algorithm remains mostly uncompromised.

Acknowledgement

Katharina Brunner is supported by the program 'Talente' by the Federal Ministry of the Republic of Austria for Climate Action, Environment, Energy, Mobility, Innovation and Technology (BMK).

Bernhard Hametner is supported by a grant of the government of Lower Austria and the European Regional Development Fund (ERDF), contract No. WST3-F-5030665/010-2021

References

[1] WHO, *Global strategy on digital health 2020-2025*. ISBN: 78-92-4-002092-4, License: CC BY-NC-SA 3.0 IGO

[2] Nahum-Shani I, Smith S, Spring B, Collins L, Witkiewitz K, Tewari A, Murphy S. Just-in-Time Adaptive Interventions (JITAIs) in Mobile Health: Key Components and Design Principles for Ongoing Health Behavior Support. In: *Ann. Beh. Med.*, 52(6):446-462, 2018. doi: 10.1007/s12160-016-9830-8.

[3] van Otterlo M, Wiering M. Reinforcement learning and markov decision processes. In: *Reinforcement Learning: State-of-the-Art*. 2012. p 3-42. doi: 10.1007/978-3-642-27645-3_1.

[4] Tewari A, Murphy S. From Ads to Interventions: Contextual Bandits in Mobile Health. In: *Mobile Health: Sensors, Analytic Methods, and Applications*. 2017. p 495-517. doi: 10.1007/978-3-319-51394-2_25

[5] Auer P, Cesa-Bianchi N, Fischer P. Finite-time Analysis of the Multiarmed Bandit Problem. In: *Mach. Learn.*, 47:235-256, 2002. doi: 10.1023/A:1013689704352

[6] Nahum-Shani I, Heckler EB, Sprujitz-Metz D. Building health behavior models to guide the development of just-in-time adaptive interventions: A pragmatic framework. In: *Health Psychol.*, 34(0):1209-1219, 2015. doi: 10.1037/hea0000306

[7] Li L, Chu W, Langford J, Schapire RE. A contextual-bandit approach to personalized news article recommendation. In: *Proceedings of the 19th International Conference on World Wide Web*; 2010 Apr; Raleigh. 661–670. doi: 10.1145/1772690.1772758

[8] Agrawal S, Golya N. Analysis of Thompson Sampling for the Multi-armed Bandit Problem. In: *Proceedings of the 25th Annual Conference on Learning Theory*; 2012 Jun; Edinburgh. JMLR. 39.1-39.26.

[9] Rocca J, Rocca B. *The exploration-exploitation trade-off: intuitions and strategies*. `https://towardsdatascience.com/the-exploration-exploitation-dilemma-f5622fbe1e82`, accessed on 2021-07-16

[10] Auer P, Using confidence bounds for exploitation-exploration rade-offs. In: *JMLR*, 3:397-422, 2002.

[11] Chapelle O, Li L. An empirical evaluation of Thompson Sampling. In: *Proceedings of the 24th International Conference on Neural Information Processing Systems*; 2011 Dec; Granada. 2249-2257.

[12] Kaufmann E, Cappe O, Garivier A. On bayesian upper confidence bounds for bandit problems. In Lawrence ND, Girolami M, editors. *Proceedings of the 15th International Conference on Artificial Intelligence and Statistics*; 2012 La Palma. 22:592-600.

[13] Agrawal S, Goyal N. Thompson sampling for contextual bandits with linear pay-offs. In Dasgupta S, McAllester D, editors. *Proceedings of the 30th International Conference on Machine Learning*; 2013 Jun; Atlanta. 28:127-135.

[14] Rabbi M, Klasnja P, Choudhury T, Tewari A, Murphy S. Optimizing mHealth interventions with a bandit. In Baumeister H, Montag C, editors. *Digital Phenotyping and Mobile Sensing: New Developments in Psychoinformatics*. Cham: Springer; 2019. 277-291. doi: 10.1007/978-3-030-31620-4.

[15] Liao P, Greenewald K, Klasnja P, Murphy S. Personalized HeartSteps: A reinforcement learning algorithm for optimizing physical activity. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*; 2020 Mar. 4(1):1-22. doi: 10.1145/3381007.

[16] Rabbi M, Aung MH, Zhang M, Choudhury T. MyBehavior: automatic personalized health feedback form user behaviors and preferences using smartphones. In: *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*; 2015 Sep. 707-718. doi: 10.1145/2750858.2805840.

[17] Paredes P, Gilad-Bachran R, Czerwinski M, Roseway A, Rowan K, Hernandez J. PopTherapy: Coping with stress in pop culture. In: *Proceedings of the 8th International Conference on Pervasive Computing Technologies for Healthcare*; 2014 May; Oldenburg. 109-117. doi: 10.4108/icst.pervasivehealth.2014.255070.

[18] Bouneffouf D, Rish I, Cecchi G, Féraud. Context attentive bandits: Contextual bandit with restricted context. In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence*; 2017 Aug; Melbourne. 1468–1475. doi: 10.24963/ijcai.2017/203

Solving ARGESIM Benchmark CP2 'Parallel and Distributed Simulation' with Open MPI/GSL and Matlab PCT – Monte Carlo and PDE Case Studies

David Jammer[1,2*], Peter Junglas[2], Sven Pawletta[1]

[1]Research Group Computational Engineering and Automation, University of Applied Sciences Wismar, Philipp-Müller-Straße 14, 23966 Wismar, Germany;
[*]*david.jammer@cea-wismar.de*

[2]PHWT-Institut, PHWT Vechta/Diepholz, Am Campus 2, 49356 Diepholz, Germany;

SNE 32(4), 2022, 211-220, DOI: 10.11128/sne.32.bncp2.10625
Received: 2022-10-19; Revised: 2022-11-10
Accepted: 2022-11-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. The ARGESIM benchmark CP2 provides three different tasks to study current technologies for the parallelization of simulation programs. The first task is the Monte Carlo study. In this study, a spring-mass system is simulated with different damping factors. The second task is a Latice Boltzmann simulation in which the flow of a fluid in a special geomentry is simulated. The third problem is a partial differential equation (PDE) describing a swinging rope, which is solved by the Method of Lines. The Monte Carlo and the PDE study are solved here, each one with two different methods: The first one applies the standard MPI message passing library together with the GNU Scientific Library, the second one uses Matlab from The MathWorks in combination with the Parallel Computing Toolbox. A special focus of this work is on the parallel processing functions provided by Matlab. The solutions are compared with each other in terms of performance and scalability. In most cases, the solutions with OpenMPI and GSL were faster than the solutions with Matlab PCT. The Matlab PCT offers many functionalities and applications to accelerate, but these usually have a poor runtime behavior.

Introduction

In simulation technology, methods to accelerate simulation were investigated in the early phases. The first benchmarks (CP1) of the SNE series dealing with this topic date back to 1994 [1] and were successfully solved and investigated with different technologies and on different platforms. This benchmark got an update (CP2) in 2007 [2] to adapt it to the increasing computing power. Unfortunately, no further solutions were submitted after this change. Since 2007, the computing power and the architecture of the hardware and software have changed a lot, so the parallel benchmarks should be brought back to life.

In this paper, two tasks of CP2 will be investigated. The tasks were implemented with two different technologies.

The first technology is the Message Passing Interface (MPI) [3] in version 4 together with the GNU Scientific Library (GSL) [4]. MPI was developed in the early 1990s and standardized in 1994. Since then MPI has been developed continuously and is still one of the standard technologies in parallel processing. MPI has been implemented by several institutes. In this paper Open MPI 4 [5] was used. GSL was developed in 1996 by M. Galassi and J. Theiler from Los Alamos National Laboratory and is currently updated and further developed. The solutions designed with it were implemented in the C language. Thus, the first solution is based on open source solutions.

The second technology is based on Matlab. In 1995 C. Moler published that The Mathworks would not be active any longer in the field of parallel processing with Matlab because of unsuccessful investigations [6]. But already in the early 1990s several open source projects had started to enable parallel processing with Matlab and similar systems. One of the first projects was developed by our research group and was presented on the Matlab Conference 1995 [7]. This development resulted in the Distributed & Parallel Toolbox [8]. In the following decade a number of similar open source projects appeared [9]. In 2004 The Mathworks adopted these developments eventually and published the first version of the Distributed Computing Toolbox [10], which became later the Parallel Computing Toolbox (PCT) [11]. The investigations in this paper are based on the PCT Version 7.4.

The computations were performed on the Seneca cluster of the PHWT-Institut. It consists of 3 nodes that communicate with each other over the high-speed InfiniBand network. Table 1 lists the data from Seneca.

Nodes	3
Cores	288
Processors	AMD Epyc 7552
Main memory	1536 GiByte
High-speed network	100 GBit/s InfiniBand
Management network	1 GBit/s Ethernet
Operating system	OpenSuse Leap 15.3
Middleware	OpenHPC
Cluster management	Warewulf
Job Scheduler	SLURM
Software	GCC 9.3.0
	GSL 2.6
	open MPI 4.1.1
	ucx 1.13.0
	libfabric 1.13.0
	hwloc 2.1.0
	Matlab R2021a

Table 1: Seneca hardware and software overview.

1 Initial Investigations

One of the most important factors in parallel processing is communication. For an application to be significantly accelerated by parallel processing, high communication performance is required. In the HPC domain, special high-speed networks are used for this purpose, which are very expensive but have great advantages over standard technologies. A common high-speed network is InfiniBand from NVIDIA (former Mellanox), which is also used in Seneca. The big advantage of InfiniBand is its low latency and high data transfer rate. To make the results of this benchmark comparable between machines with different architectures, we will initially provide some results of corresponding measurements.

Figure 1 displays the round trip time and the transmission rate between two cores of different nodes as a function of the packet size.

If the packet size approaches 0, then this corresponds approximately to the latency. This latency is approx. 3 μs for Seneca. The maximum transmission rate is approx. 96 GBit/s.

Another communication medium is the main memory. Since the multicore technology has been strongly enhanced for several years and the performance continues to increase, it is also used massively in the HPC domain. The AMD processors used are based on the NUMA (Non-Uniform Memory Access) architecture.

Two investigations can be carried out here, which measure different aspects of the communication via the memory inside one node: Firstly, the exclusive communication between two cores (cf. Figure 2), and secondly, the simultaneous communication of several cores. Figure 3 shows the corresponding transfer rate as a function of the number of cores.

In Figure 2, it is noticeable that a maximum of approx. 200 GBit/s is at a packet size of 128 KiByte. For larger packets, the transfer rate is 100 GBit/s. The latency (packet size against 0) is approx. 0.4 μs.

The transfer rate shown in Figure 3 reaches a maximum of approx. 150 GiByte/s. In this measurement, 1 GiByte was copied back and forth 500 times per core.

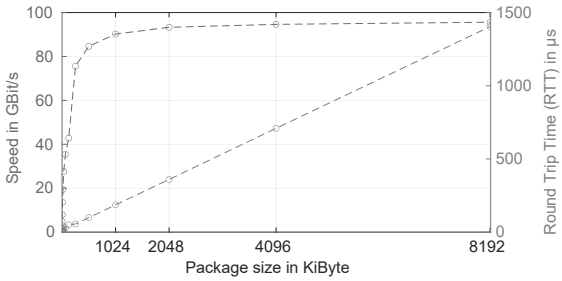


Figure 1: Communication between two nodes via InfiniBand.

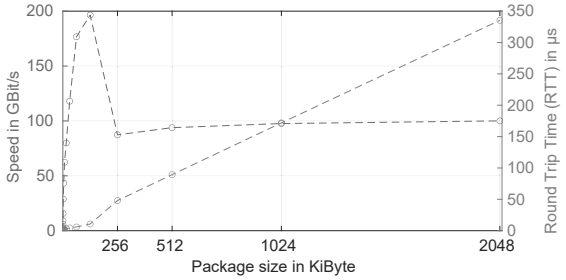


Figure 2: Communication between two cores via main memory.

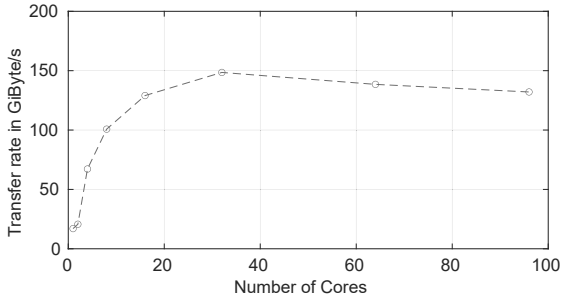


Figure 3: Memory bandwidth depending on the number of cores.

In a classic UMA (Uniform Memory Access) architecture, the transfer rate would remain constant because the instances share the memory channel to the main memory. However, since a NUMA architecture has several memory channels, the transfer rate increases because more memory channels are used.

2 Monte Carlo Study

The first benchmark is a Monte Carlo study. In this benchmark, the behaviour of a spring-mass system with different damping factors has to be calculated. The spring-mass system is described by the usual differential equation:

$$m\ddot{x} + d\dot{x} + kx = 0$$

with the parameters:

$$x(0) = 0, \dot{x}(0) = 0.1, k = 9000, m = 450$$

The damping factor d is randomly selected using a uniform distribution with the range [800,1200].

In [2] $nReps$ = 1000 simulations with a step size of h = 0.01 in the period from 0 to 2 are required. The mean value of $x(t)$ is then to be calculated from the simulations. Since the computing power of computers has increased significantly since the publication of [2], we have increased the load of the task. For the following solutions, $nReps$ = 10,000,000 simulations were performed with a step size of h = 0.001.

2.1 Open MPI with GSL

This section presents the solution using Open Message Passing Interface (MPI) [5] and the GNU Scientific Library (GSL) [4]. The program is written in the C programming language. The differential equation is transformed into the usual first order form:

$$\dot{y}_1 = y_2$$

$$\dot{y}_2 = -\frac{d}{m}y_2 - \frac{k}{m}y_1$$

and then solved with an RK4 solver with fixed step size. GSL provides all necessary functions and data structures for this task. According to [12], the development of a parallel solution, starting from the entire problem, consists of the steps partitioning, communication, agglomeration, and mapping. The partitioning of the Monte Carlo study results in the following tasks: $nReps$ simulations, calculation of the mean value and the storage of the results. The communication includes the transfer of the results of the $nReps$ simulations to the averaging operation and from there to the storage of the results. The aim of the subsequent agglomeration step is to reduce communication and combine tasks. Here, several simulations per process and the computation of corresponding partial sum vectors are combined. This reduces the communication per process to the transmission of the partial sum. Another part is the addition of all partial sum vectors, the averaging and the storage. The addition of all partial sum vectors is done collectively via a Reduce function and the averaging and storage is combined as one task and assigned to an arbitrary process. The mapping is done automatically by the middleware and the operating system.

The solution was designed as SPMD (Single-Program Multiple-Data) as shown in Figure 4. Since all simulations need the same amount of computation, a load balancing scheme is unnecessary.

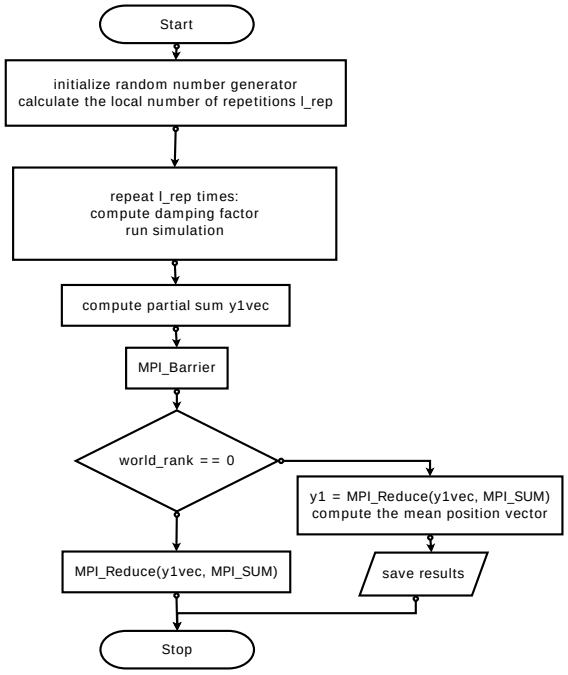


Figure 4: Program flowchart of the monte carlo study with OpenMPI.

Therefore, the number l_{rep} of local repetitions can be computed beforehand by distributing the total number n uniformly among the tasks. Then each MPI process calculates the needed random numbers and performs its simulations. The seed of each local random number generator is initialized with the ID of the MPI process to get independent random numbers for each process. The position values y_1 are added locally after each simulation, thus only one vector has to be transferred afterwards. After all processes have finished their computations, the addition of the partial sum vectors is done collectively by `MPI_Reduce`. Now, the MPI process with rank 0 contains the result of the addition and only has to calculate the average and finally store the data.

2.2 Matlab PCT

The PCT can be used for parallel programming on a local multi-core machine or on a cluster like Seneca. To use PCT, one first creates a parallel pool with a given number of processes, called "workers". On a cluster, Matlab always requires an additional worker as the top instance. This is important, because this worker also needs a license. PCT defines several different models for parallel programming.

For the Monte Carlo study three technologies have been used: `parfor`, `spmd` and `parsim`.

The solution with parfor is very easy to implement, because the decomposition is performed implicitely. Parfor works like a normal for-loop with the only difference that the iterations of the for-loop are distributed to the workers. Unfortunately, the PCT documentation doesn't describe the scheduling strategy of the parfor loop, it simply states that the iterations are done in non-deterministic order [11]. However, it is not important for this Monte Carlo study, since the computational effort of the simulations is always the same. A parfor loop is only possible if the iterations are completely independent from each other. In our case, this condition is obviously fulfilled. A vector D with $nReps$ random numbers is calculated in the sequential section before the parfor loop. In the parfor section, only the `ode45` function is called and the sum is formed as in Listing 1.

Listing 1: Matlab parfor-loop

```
parfor i=1:nReps
  [tout, yout] = ode45(@(t,y)
                       ode(t,y,K, D(i),M),
                       tvec,y0);
  ysum = ysum + yout(:,1);
end
```

The communication happens here implicitly by the calculation of the sum using the variable *ysum*, which was defined in the sequential section and can be used by all workers. After the parfor loop only the mean value remains to be computed.

Another solution was implemented and investigated with `spmd`. The `spmd` function creates a parallel section that is executed by each worker. The number of workers is given by the size of the parallel pool and can be determined in the `spmd` section via the variable `numlabs`. The index of a worker is stored in the variable `labindex`. Data can be sent explicitely between workers using the functions `labSend` and `labReceive`. For synchronization the function `labBarrier` is available. The solution with `spmd` is similar to the solution with Open MPI from section 2.1. In the `spmd` section, the local number of repetitions l_{rep} is calculated and then the simulations are performed. A special challenge is the handling of anonymous functions, because they cannot be defined in an `spmd` section. But they are needed to modify the parameters of the differential equation.

The easiest way to realize this, is to create a function outside the spmd section, which contains the ode45 call and the definition of the anonymous function. This is shown in Listing 2.

Listing 2: Matlab ode45 call with anonymous for the spmd section

```
function [t,y] = odecall(tvec,y0,K,D,M)
  [t, y] = ode45(@(t,y) ode(t,y,K,D,M),
                 tvec, y0);
end
```

Similar to section 2.1 the position values are added and the sum is calculated by the general reduction function `B = gop(fcn,A,destination)`, where `@plus` is used for `fcn` to achieve a collaborative summation.

The third solution was realized with Matlab/Simulink and parsim. For this purpose, the spring-mass system must be modeled by a signal flowchart in Simulink (cf. Figure 5). Afterwards, only

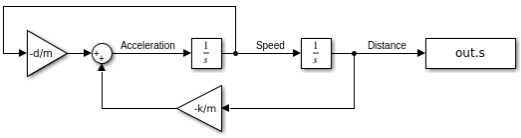


Figure 5: Spring-mass system as signal flowchart.

the input data must be defined as Simulink simulation input objects and then the simulations can be started with parsim. After the simulations are finished, the mean value can be calculated from the results. This solution is the simplest of all, but it must be noted that all results are available before averaging, which can lead to a high memory overhead.

2.3 Comparison of solutions

All approaches resulted in the same solution as shown in Figure 6.

Figure 7 shows the runtime and the speedup results of the Monte Carlo study. Apart from the parsim method, which has no speedup at all, the other implementations show a significant speedup. The runtimes of the two Matlab solutions are almost identical and have a significantly higher runtime than the C implementation. The transition from one node to two nodes is interesting: The Matlab implementations show a jump

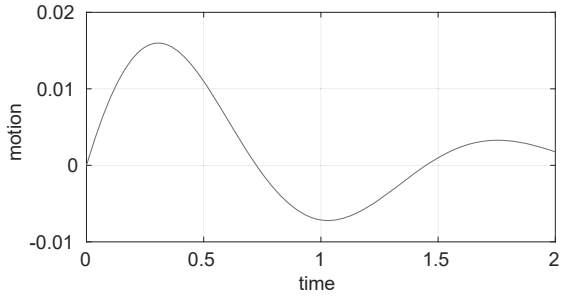


Figure 6: Plot of the mean motion.

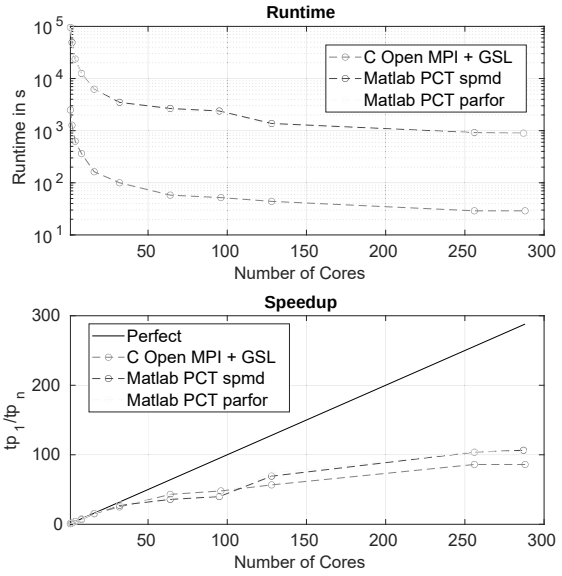


Figure 7: Runtime and speedup history of the monte carlo implementations.

in speedup in that area, whereas the C implementation does not have this jump.

The solution with Matlab and parsim did not result in any significant acceleration (cf. Figure 8). Also, the number of simulations had to be reduced significantly to cope with the high memory requirements of this solution.

Another interesting investigation in a Monte Carlo study is the scaleup. The execution time is constant in a scaleup approach, so only the spmd method was investigated in Matlab. In an spmd section, the maximum execution time can be implemented directly. Here, a time of 60 seconds was specified in which the simulations are carried out. Figure 9 then shows how many simulations are calculated depending on the number of cores used.

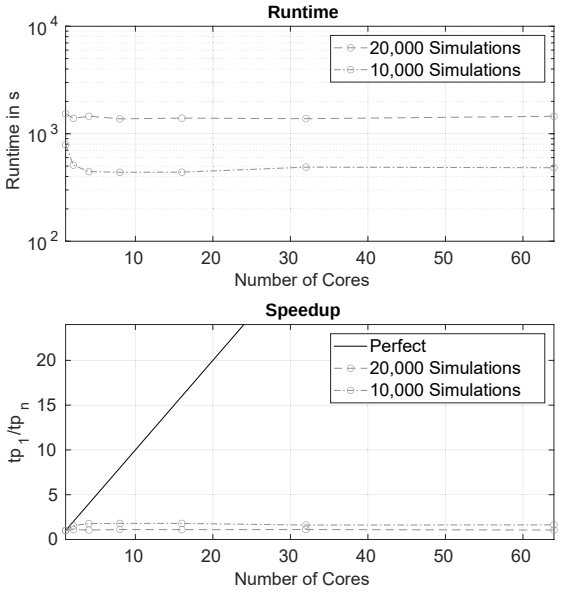


Figure 8: Runtime and speedup history of the monte carlo parsim implementation.

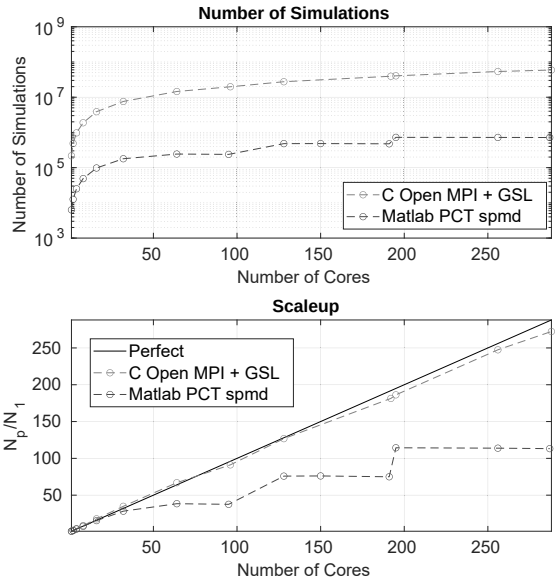


Figure 9: Scaleup history of the monte carlo implementations.

The C implementation shows an almost linear behavior, whereas the Matlab/spmd implementation behaves strangely: The transitions from one (96 cores) to two (192 cores) nodes and from two (192 cores) to three (288 cores) nodes are striking. In these transitions the number of simulations increases abruptly.

In comparison, the C implementation clearly shows the best results, as was expected. Nevertheless, the Matlab implementations can provide reasonable speedup with little programming effort.

3 Partial Differential Equation Case Study

The second benchmark investigated in this article is the solution of a partial differential equation describing a swinging rope:

$$\frac{\partial^2 u}{\partial x^2} = \frac{1}{v^2}\frac{\partial u}{\partial t^2}$$

As suggested in the benchmark, the equation will be solved with the method of lines. For this purpose, the left side of the differential equation is replaced by a central difference quotient of 2nd order:

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1}(t) - 2u_i(t) + u_{i-1}(t)}{(\frac{L}{N})^2}$$

Here L corresponds to the length of the rope, N to the number of equidistant intervals used for space discretization and i to the location. Applying this method, a 2nd order differential equation is obtained for each location point:

$$\ddot{u}_i = \frac{v^2}{k^2}(u_{i+1} - 2u_i + u_{i-1}), \qquad i = 1,...,N-1$$

with

$$k := L/N$$

The initial values are:

$$u_i(0) = 2\frac{h}{N}i, \qquad i = 1,...,\frac{N}{2}$$

$$u_i(0) = 2h(1 - \frac{i}{N}), \qquad i = \frac{N}{2},...,N-1$$

and

$$\dot{u}_i(0) = 0, \qquad i = 1,...,N-1$$

The boundary values are given by

$$u_0(t) = u_N(t) = 0, \qquad t \in [0, t_{end}]$$

The parameter values are given as $\nu = 0.06$, $L = 0.5$, $h = 0.05$ and $t_{end} = 10$. The benchmark specifies the space discretization as $N = 500$ and the size of time steps as $dt = 0.01$. In the following, other values will be used to adapt to the increase in computing power. As a solution, the time evolution of the amplitude at the points $x = \frac{3}{4}L$ and $x = \frac{1}{2}L$ is to be presented, as well as the space evolution at the points in time $t = 5$ and $t = 8$ and a surface plot, showing the complete solution $u(x,t)$.

3.1 Open MPI with GSL

Using the design method for parallel programs described in section 2.1 results in a very fine grained description: The main tasks are the computations of one time step at one position. This leads to a huge amount of communication, which will be reduced drastically after the proper agglomeration step. Since the resulting strategy is fairly standard, we will skip these details and immediately describe the overall parallelization strategy.

The basic idea is to distribute contiguous uniformly sized sections of the rope to the tasks, in the order of the taskIds (cf. Figure 10). Note that the local sections of the first and last taskIds include the endpoints u_0 and u_N, which contain the fixed boundary values. The algorithm consists of a time loop, where each iteration starts with an exchange of the necessary boundary values, followed by one step of an RK4 ODE solver. A barrier between steps synchronizes the tasks to guarantee the correct internal boundary values. As part of the steps all necessary results are collected by the task with taskId 0 and stored.

Figure 10: Decomposition of space points.

To organize the communication and to simplify the local computations so called "ghost points" ([13]) are used (cf. Figure 11): The local u arrays are extended by one additional point at each end (or only one end for the first and last task). In the communication phase the boundary values are stored here, so that all necessary values are readily available at the computation phase.

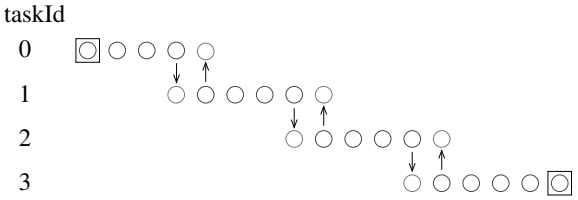


Figure 11: Decomposition including ghost points.

3.2 Matlab PCT

Three solution approaches were investigated using Matlab PCT. The first one ("loop-based") closely mimics the MPI version adopting the `spmd` environment and the `labSend`, `labReceive`, `labindex` and `labBarrier` functions. As in MPI the definition of the ODE is done in a loop over the space points. For the tedious task of mapping between the global and local indices – prone to typical one-off errors – Matlab PCT supplies the function `codistributor1d`, which provides all necessary details.

For the second approach ("matrix-based") the variables u_i are combined in a vector u and the ODE is rewritten in matrix-vector form as

$$\ddot{u} = Au$$

with

$$A = \frac{v^2}{k^2} \cdot \begin{pmatrix} -2 & 1 & 0 & 0 & \cdots \\ 1 & -2 & 1 & 0 & \cdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \cdots & 0 & 1 & -2 & 1 \\ \cdots & 0 & 0 & 1 & -2 \end{pmatrix}$$

The local parts of A are computed in an initial phase. The parallelization follows the pattern of the loop-based version, only the internal loop over points is replaced by a matrix-vector multiplication.

The third approach ("codistributed") applies codistributed arrays, a very convenient tool defined in Matlab PCT. A codistributed array basically is an array that is distributed to the workers, but every task can still access each element with the usual (global) index. If the corresponding element is not locally available, it is sent automatically from the hosting worker to the accessing one. Auxiliary functions are supplied to find the local indices, but they are not needed here: Several Matlab functions, such as the matrix-multiplication, automatically cope with codistributed arrays, so that a simple call by all workers is sufficient.

Of course, the crucial point here is the distribution scheme. Matlab PCT supplies the usual equal-partitioning along rows or columns and a block-cyclic twodimensional partitioning.

Surprisingly, the loop-based approach reached the shortest runtime, while the matrix-based approach was slower. The codistributed approach is very elegant from a programming point of view, since the distribution and communication parts are done implicitly, but it is extremely slow compared to the other solutions. Table 2 shows the runtimes of the implementations in comparison. For the further investigations only the loop-based approach has been used.

solution approach	runtime [s]
loop-based	4.23
matrix-based	5.77
codistributed	359.85

Table 2: Running time of the three solutions (nWorker=8, dt=0.001, N=1000).

3.3 Comparison of solutions

All implementations returned the same result. Figure 12 shows the excitation over time and space.

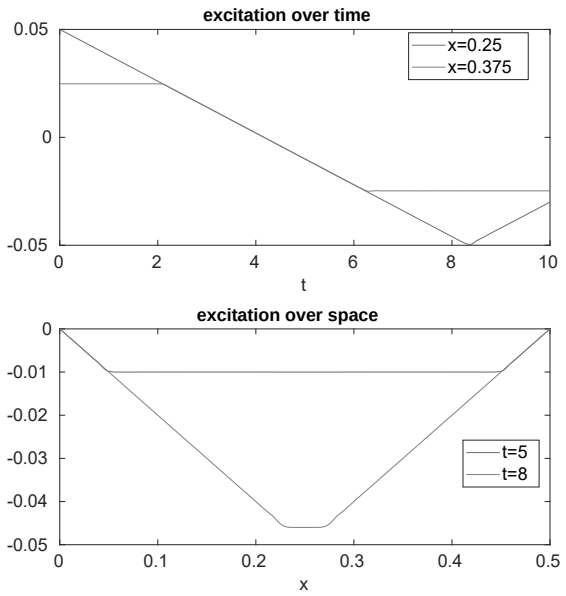


Figure 12: Solution of the PDE, excitation over time and space.

Figure 13 shows the required surface plot.

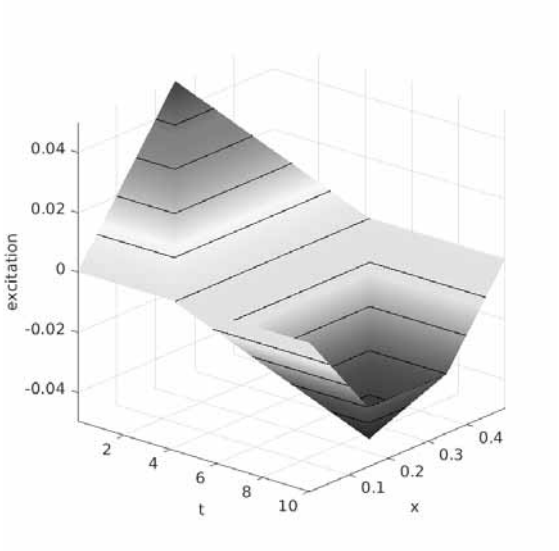


Figure 13: Surface plot for the swinging rope.

Since for the surfaceplot all data over time and space must be stored, this task does not scale well to large values of N. The result files become too large at higher space and time resolutions. In order to increase the computational effort nevertheless, the collection of all data was omitted for the following measurements. Only the excitation over time at the locations $x = \frac{3}{4}L$ and $x = \frac{1}{2}L$ and the excitation over space at the times $t = 5$ and $t = 8$ were collected and stored.

The runtime of the two implementations Open-MPI/GSL and Matlab PCT/loop-based was investigated depending on the number of processes. For comparison, runs with $N = 200,000$ and $N = 1,000,000$ have been studied. The results of the speedup analysis are shown in Figure 14 and 15. A significant speedup was reached in all scenarios. The C implementation with Open MPI/GSL reached the lowest runtimes and also the highest speedups. The runtime was reduced from about 20 minutes (N=200,000) to about 1 minute and from about 14 hours (N=1,000,000) to about 6 minutes.

This corresponds to a maximum speedup of about 20 for N=200,000 and about 140 for N=1,000,000. The Matlab PCT/loop-based solution reduced the runtime from 30 minutes to 2 minutes (N=200,000) and from about 10 hours (N=1,000,000) to 25 minutes, respectively. This is equivalent to a speedup of 15 for N=200,000 and 23 for N=1,000,000.

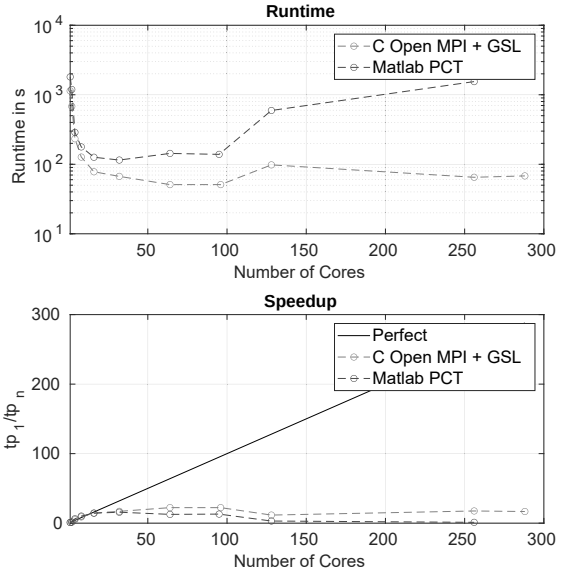


Figure 14: Runtime and speedup history of the PDE implementations with N=200,000, dt=0.00005 and t_{end}=10.

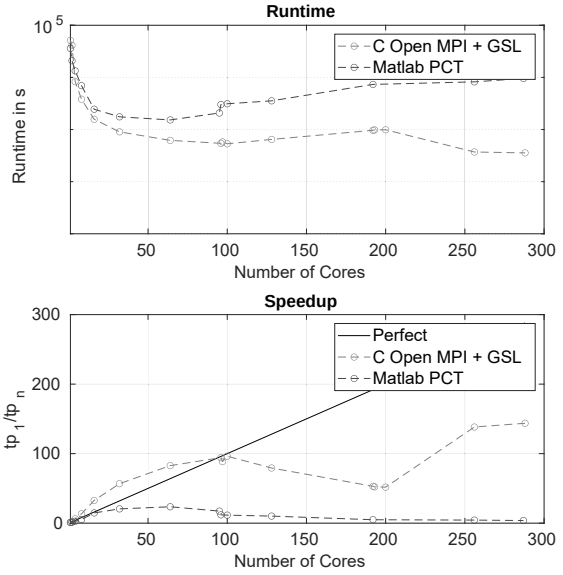


Figure 15: Runtime and speedup history of the PDE implementations with N=1,000,000, dt=0.00001 and t_{end}=10.

The bad speedup behaviour for N=200,000 indicates that this problem is still too small for the hardware, whereas for N=1,000,000 the MPI/GSL version shows reasonable speedups even for large core numbers. The superlinear speedup for one cluster probably is due to a better use of the memory lines and caches. Interestingly, Matlab PCT is faster than MPI/GSL for one core, but generally has much less speedup and can't put more than one cluster node with 96 cores to good use.

4 Conclusion

In this article, two solutions for each of two tasks of the ARGESIM CP2 benchmark have been presented and compared with each other. Both tasks are very lightweight for today's computer systems, the sequential Monte Carlo study takes less than 1 second with the original parameters. Therefore, the tasks have been scaled up considerably. A revision of the CP2 benchmark should be scalable and allow for parameters that lead to runtimes of 30 minutes or more.

As expected, the implementations in C with Open MPI/GSL generally reached shorter runtimes and reasonable speedups for up to more than 100 cores, especially for large problem sizes. The basic Matlab PCT solutions using `parfor` or `spmd` showed good speedups for up to over 100 cores in the Monte Carlo study, while in the PDE example good speedups could only be reached for up to 30 cores. Using high-level methods like `parsim` or codistributed arrays produced elegant looking programs, but they could not compete at all due to their huge execution times.

Another problem is the insufficient documentation of Matlab PCT: Its authors try to spare the reader most internal details, such as the load-balancing scheme of `parfor` or the exact behaviour of codistribution-aware functions, which are important for good parallelization strategies.

In a subsequent work the third task of the CP2 benchmark, the Lattice Boltzmann simulation, will be solved, again using implementations in OpenMPI and Matlab PCT.

References

[1] Breitenecker F, Husinsky I, Schuster G. Comparison of Parallel Simulation Techniques. *SNE Simulation News Europe*. 1994;4(10):21–22. ISSN 0929-2268.

[2] Breitenecker F, Höfinger G, Pawletta T, Pawletta S, Fink R. ARGESIM Benchmark on Parallel and Distributed Simulation. *SNE Simulation News Europe*. 2007; 17(1):53–56. ISSN 0929-2268.

[3] MPI: A Message-Passing Interface Standard. URL `https://www.mpi-forum.org/docs/mpi-4.0/mpi40-report.pdf`

[4] Galassi M, et al. *GNU Scientific Library Reference Manual*. 3rd ed. URL `https://www.gnu.org/software/gsl/`

[5] The Open MPI Project. URL `https://www.open-mpi.org/`

[6] Moler C. Why there isn't a parallel MATLAB. In: *Matlab News and Notes*. Spring. 1995; p. 12.

[7] Pawletta S, Pawletta T, Drewelow W. A MATLAB toolbox for distributed and parallel processing. In: *2nd International MATLAB Conference*. Cambridge. 1995; .

[8] Pawletta S, Pawletta T, Drewelow W, Dünow P. *Distributed and Parallel Application Toolbox for Use with MATLAB: User's Guide and Reference Manual*, version 1.3 ed. 1996.

[9] Fink R, Pawletta S, Pawletta T, Lampe B. SCE based Parallel Processing and Applications in Simulation. *SNE Simulation News Europe - Special Issue on Parallel and Distributed Simulation Methods and Environments*. 2006;16(2):37–50. ISSN 0929-2268.

[10] The Mathworks Inc. *Distributed Computing Toolbox for Use with MATLAB: User's Guide*, version 1 ed. 2004.

[11] The Mathworks Inc. *MATLAB Parallel Computing Toolbox: User's Guide*, version 7.4 ed. 2021.

[12] Foster I. *Designing and Building Parallel Programs - Concepts and Tools for Parallel Software Engineering*. Amsterdam: Addison-Wesley. 1995.

[13] Gropp W, Lusk E, Skjellum A. *Using MPI, third edition - Portable Parallel Programming with the Message-Passing Interface*. Cambridge: MIT Press. 2014.

Compartment Modeling of Overweight in Toddler Age: Modeling and Simulating a Diets Effect with COPASI

Priska Bauerstätter[1,2*]

[1]Austrian Institute of Technology (AIT), Center for Health and Bioresources, Giefinggasse 4, 1210, Vienna, Austria
*priska.bauerstaetter@ait.ac.at

[2]TU Wien, Inst. of Analysis and Scientific Computation, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria

SNE 32(4), 2022, 221-224, DOI: 10.11128/sne.32.sn.10626
Received: 2021-10-10; Revised: 2022-07-20
Accepted: 2022-08-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. Since overweight in toddler age has become an increasing problem in the last years it is an extensive and important topic. To counteract this, new diets are constantly being launched. But how effective are they really? To answer this question we present a system modeling approach for simulating the impact of an arbitrary diet. More precisely we design a simple model for the different body weight states and their transitions, model it in COPASI and do some simulations to discuss the impact of an arbitrary diet.

Introduction

Since overweight has become a problem in the last years it is an extensive and important topic. Increasingly it is already noticed in childhood too. In addition to limited quality of life, it also has enormous health consequences, such as coronary heart disease, type 2 diabetes mellitus and hypertension. Furthermore all these diseases cause high costs. Studies show that obesity is becoming more common in early childhood. This leads to many personal disadvantages, such as the impairment of the quality of life of the affected people. In addition, overweight causes public costs too, because overweight people have higher disease and accidents probabilities as well as higher unemployment rates.

Motivation. Modeling and simulating complex problems can answer many questions. Depending on the question, an abstract model of the system is needed. In order to answer such a question COPASI takes the abstract model as input and simulates its behaviour.

Thereby, it can be checked if the expected behaviour is observed, analysed how the desired behaviour can be reached or improved, as well as some specific questions according to the modelled problem can be answered.

Related Work. A number of related approaches in the area of modeling and simulating populations is available. For example, Levy et al. [2] compared different simulation models of obesity. In contrast to our approach, they discuss much more complex models. To the best of our knowledge, there is no related work on modeling this simulation model within COPASI.

Contribution. We define a simulation model and implement it by using COPASI. Furthermore, we analyse the impact of a diet by comparing two models (one with and one without diet states). With this simulation, we can determine if the diet has a positive effect and which arrangements should be done to improve people's bodyweight conditions.

1 COPASI Modelling & Simulation

COPASI is a very powerful system simulator for reactions that convert a set of (any kind of) species into another set of species.

Following [1] and the COPASI webpage, in COPASI models each species is located in a compartment, which is a physical location with a size (volume, area, etc.). This maps directly to biochemical reaction networks, but can also represent other types of processes (for example, the species could be cell types). COPASI automatically converts the reaction network to a set of differential equations or to a system of stochastic reaction events.

Furthermore, COPASI models can have an almost unlimited number of species, reactions, and compartments, they can include arbitrary discrete events (also for model

change), and arbitrary differential equations can be added explicitly and can be mapped to species, compartments, or generic variables. Compartments can have variable sizes, and for the reaction rates a large set of predefined kinetic functions are provided or can be defined by the user. COPASI not only comes along with a graphical modelling environment, it offers also data visualisation and result visualisation in 2D, in 3D, with animation, chart representation and others.

The key tool in COPASI is the multifaceted model analysis, not only in the time domain as many system simulators. COPASI provides (from [1]):

- stoichiometric analysis of reaction network, with mass conservation analysis and elementary flux modes,
- optimization of arbitrary components of the model,
- parameter estimation using a range of diverse optimization algorithms,
- local sensitivity analysis,
- metabolic control analysis,
- time scale separation analysis,
- analysis of stochasticity (linear noise approximation),
- cross section analysis (characterisation of nonlinear dynamics, as oscillations and chaos),
- and Lyapunov exponent calculation for chaos detection.

For these analysis methods, COPASI makes use of the following algorithms:

- LSODA for ODEs,
- RADAU5 for stiff ODEs,
- Stochastic Runge-Kutta (RI5) for stochastic ODEs,
- Gillespie's direct method for exact stochastic kinetics,
- Gibson & Bruck's version of Gillespie's algorithm for exact stochastic kinetics,
- τ-leap algorithm and adaptive SSA/τ-leap algorithm for faster stochastic kinetics,
- hybrid Runge-Kutta/SSA and hybrid LSODA/SSA algorithms for mixed stochastic kinetics and ODEs,
- and hybrid Runge-Kutta/SSA with arbitrary partition of reactions between stochastic kinetics and differential equations.

For the compartment modelling of overweight, we only use very basic features of COPASI: compartments (group with certain overweight status), with flow due to linear kinetics (1st order kinetics).

2 Method

In this section, we show how we designed the model and how this model can be applied for simulating the system behaviour. This approach is demonstrated by a simple example of a body weight condition model for toddlers.

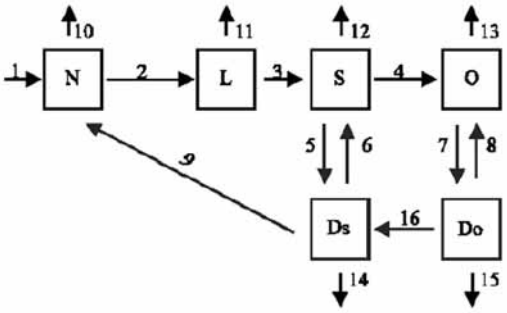


Figure 1: Model of typical body weight conditions and their diet states.

Model. First we designed the model in Figure 1. At the beginning every infant has normal body weight (state N). If the infant has unhealthy eating habits, its state changes to L. With bad eating habits, the probability for weight increase is high. Therefore the affected toddler changes to state S and if its weight is further increasing it gets obese (state O). If an infant improves its eating habits, it is on diet (DS for overweight toddlers on diet and Do for obese toddlers on diet) and can change back to its state before, because of weight loss. Furthermore, every state has his own transition for ending up within the state (transitions 10 to 15). Clearly, for getting a better diet effect, the rates of the transitions to normal weight should be maximized and minimized for staying on diet or stopping it without weight loss.

More mathematically the model above yields one differential equation per state. E.g. the differential equation for state S is defined as:

$$\frac{dS(t)}{dt} = t_3 L + t_6 Ds(t) - (t_4 + t_5 + t_{12})S(t)$$

For the parameter S there are positive rates t_3 and t_6 for the incoming transitions, depending on the population size of state L and Ds, and negative for the outgoing ones (t_4, t_5, and t_{12}), depending on S. Analogous the five remaining differential equations could be built.

Using COPASI, all these ODEs are generated automatically via the GUI using the model as in Figure 1.

For simulation, we used classic time domain analysis by COPASI's ODE solvers (which are LSODA or

LSOAR [3] by default), and we used the steady state calculation algorithm method (for more details have a look at the user manual on the COPASI website [4]).

It is to be noted, that the model is a linear one, and that the above ODE for state S is equivalent to a transfer function model with

$$\mathbf{S}(s) = \frac{1}{s + (t_4 + t_5 + t_{12})} \cdot (t_3 \cdot \mathbf{L}(s) + t_6 \cdot \mathbf{Ds}(s))$$

Here $\mathbf{S}(s)$, $\mathbf{L}(s)$, and $\mathbf{Ds}(s)$ are the Laplace trasformations of $S(t)$, $L(t)$, and $Ds(t)$.

This equivalence – compart model with 1st order kinetics and 1st order transfer function allows the interpretation of the model in Figure 1 as (linear) network of 1st order transfer functions. Obviously, in this case also linear time domain analysis would have been sufficient.

Data. We got empirical data from fictitious hospital in Vienna of 1187 infants and their body weight progress. Based on that, the initial values for the transition probabilities are set as shown in Table 1.

var	value	var	value	var	value	var	value
t_1	6	t_2	0.02	t_3	0.0089	t_4	0.0029
t_5	0.0407	t_6	0.1274	t_7	0.0444	t_8	0.1598
t_9	0.0278	t_{10}	0.0006	t_{11}	0.0006	t_{10}	0.0006
t_{13}	0.0006	t_{14}	0.0006	t_{15}	0.0006	t_{14}	0.0042

Table 1: Calculated transition probabilities from log data (commercially rounded to four decimal places).

To see the course over time, we simulated 8000 time steps with interval size 1. Therefore, the integration and output intervals are from 1 to 8000. The initial values at time 0 are listed in Table 2.

Model with diet		Model without diet	
$N(0)$	255	$N(0)$	255
$L(0)$	226	$L(0)$	226
$S(0)$	273	$S(0)$	546
$O(0)$	95	$O(0)$	190
$Ds(0)$	273	$Ds(0)$	–
$Do(0)$	95	$Do(0)$	–
Sum	*1217*	*Sum*	*1217*

Table 2: Initial values at time 0 for both models. The model with the two diet states Ds and Do on the left and the model without diet states on the right.

Result. Figure 2 shows the simulation according to the input values. The numbers after 8000 time steps are almost the steady state numbers.

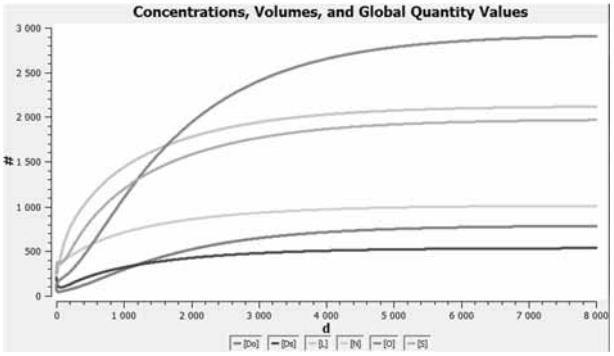


Figure 2: Course of the diet model over 8000 time steps.

To see the diets effect, analogous to our described model we built a second model without the two diet states Ds and Do. Hence this model is exactly the same, all parameters and methods are the same but without overweight and obese toddlers on diet. The states and transitions are sketched in Figure 3, which is a reduced model of the model with diet states (see Figure 1).

To keep the comparability we added the remaining initial values of Ds and Do to the initial values for the overweight and obese toddlers (see Table 2).

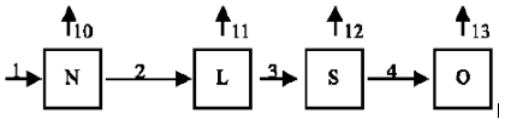


Figure 3: Model of typical body weight conditions without diet states.

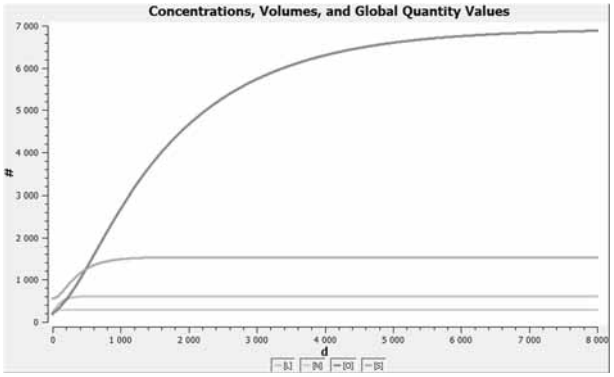


Figure 4: Time Course of the model without diet over 8000 time steps.

By comparing the two COPASI outputs (see Figure 2 and Figure 4) we can see that the diet led to a reduction of the adipose infants by more than 50% and while the overweight toddlers have increased slightly, the normal weights and the normal weights with bad eating habits increased too.

As result, by comparing the two models in their steady states we can conclude that the introduction of the diet was successful. Therefore, the diet should be applied to infants. Note that in reality, there are further reasons for overweight and weight loss in toddler age. To keep it simple we omitted them.

3 Conclusion

We have demonstrated how body weight conditions could be modelled and simulated with COPASI, and analysed if a given diet has a positive effect on body weight conditions of infants.

More general, with such a model it could be analysed how the parameters and variables affect the overall behaviour of a body weight condition system and answered different questions, e.g. which parameters or variables have the greatest effect on desired outcomes or how a specific variable influences the system flow.

The selected model implementation is very simplified, but it is good enough to show how a system could be modelled, simulated and analysed.

A big advantage of this 'reduced' compartment modeling is, that reality is sufficiently related to the modelled variables. Therefore it is not really important having all observed variables explicitly in the model, which simplifies the analysis a lot.

Acknowledgement. This simulation study was integrated into a project work for the TU Wien lecture 'Control Models in Physiology and Medicine', supervised by Felix Breitenecker. The equivalence of the ODEs for the overweight states with transfer function models is used as basis for a linear time domain analysis of a network of transfer functions and of an overall state space model, to be published in a following SNE publication as *Educational Note*.

References

[1] Hoops S, Sahle S, Gauges R, Lee C, Pahle J, Simus N, et al. COPASI—a COmplex PAthway SImulator. *Bioinformatics.* (2006) 22:3067–74. DOI 10.1093/bioinformatics/btl485

[2] Levy DT, Mabry PL, Wang YC, Gortmaker S, Huang TT, Marsh T, Moodie M, Swinburn B. Simulation models of obesity: a review of the literature and implications for research and policy. Obes Rev. 2011 May; 12(5):378- 94. DOI 10.1111/j.1467-789X.2010.00804.x.

[3] Petzold L. *Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations.* SIAM J. Sci. Stat. Comput. 4. 136 - 148. 1983.

[4] COPASI user manual (2021, Feb. 28). Retrieved from http://copasi.org/Support/User_Manual/

Methods for Integrated Simulation 10 Concepts to Integrate

Niki Popper[1*,2], Martin Bicher[1], Felix Breitenecker[3], Barbara Glock[1,4], Irene Hafner[1], Miguel Mujica Mota[5], Gašper Mušič[6], Claire Rippinger[1], Matthias Rössler[1], Günter Schneckenreither[1], Christoph Urach[1,4], Matthias Wastian[1,4], Günther Zauner[1,4], Melanie Zechmeister[1,4]

[1]Research Unit of Data Science, TU Wien, Favoritenstraße 9–11, 1040 Vienna, Austria; *nikolas.popper@tuwien.ac.at*
[2]Research Unit of Computational Statistics, TU Wien, Wiedner Hauptstraße 8–10, 1040 Vienna; Austria
[3]Inst. of Analysis and Scientific Computing, TU Wien, Wiedner Hauptstraße 8–10, 1040 Vienna; Austria
[4] dwh Simulation Services, Neustiftgasse 57-59, 1070 Vienna;
[5] Amsterdam University of Applied Sciences, Aviation Academy, 1097 DZ Amsterdam, The Netherlands
[6] Faculty of Electrical Engineering, University of Ljubljana, Tržaška 25, 1000 Ljubljana, Slovenia

SNE 32(4), 2022, 225-236, DOI: 10.11128/sne.32.on.10627
Received: 2022-10-15; Revised: 2022-12-10
Accepted: 2022-12-15
SNE - Simulation Notes Europe, ARGESIM Publisher Vienna,
ISSN Print 2305-9974, Online 2306-0271, www.sne-journal.org

Abstract. This note summarises the current status of the work of EUROSIM's and ASIM's Technical Committees "Data Driven System Simulation" - with main emphasis on Big Data integration in simulation. This overview suggests ten developed concepts and methods, which should be considered, implemented and documented in modern simulation studies with Big Data.

Motivation

Diversity and heterogeneity of man-made systems is rapidly increasing and with it the costs spent on them. For a long time, these systems supposedly worked well. Currently, however, we are seeing a number of challenges, be it in the area of energy, mobility, logistics or health systems. Here, costs are rising, there are supply bottlenecks and, above all, the lack of resilience of the systems, i.e. the adaptation to changing conditions, is a major challenge.

Measuring efficiency and effectiveness of such systems is becoming increasingly complex, but is urgently needed. The development of new methods, models and simulations is necessary to support analysis, planning and control. Especially the possibility to calculate "what if" scenarios is an absolute necessity in order to be able to react to changing framework conditions.

The heterogeneity of the systems requires the possibility of integrating different modelling concepts in order to be able to depict the systems in sufficient detail. In addition, the quantity and quality of the available data are increasing strongly and thus facilitate the descrip-tion and analysis of such systems.

On the other hand, this increases the effort to parameterise, calibrate and validate the models and simulations. Bringing together the necessary technologies is thus itself an enormous challenge.

1 Outline

Data-based Demographic models have to be combined with models for the spread of diseases. Dynamic modelling concepts must be parameterised with dynamically changing data sets from various sources.

For system simulation an important aspect is the possibility to implement changes inside the system, like interven-tions within the computer model, and to analyse their effects. As a recent example see Covid-19 Modelling at TU Wien [1]).

Based on the concepts of equations, networks, algorithms and the causal understanding of the world, modelling and simulation have reached a high level in describing systems and processes, e.g. complex technical systems, ecological systems, production and logistics processes or socio-economic systems such as the healthcare system.

On the other hand, Big Data based on sensors and computations to measure our world has gained outstanding importance. Today, there is a range of technologies for building, monitoring and evaluating it. A multitude of activities can be observed in research, development, politics and the media.

Nevertheless, interfaces and methods for linking these technologies need to be intensified. In particular, complex socio-technical systems that link technologies and people should serve the goals of citizens at different levels, from a citizen's personal goals, e.g. in terms of work or mobility, to the management of health care by politicians and decision-makers. A prerequisite for this is the analysis of actual data, the prediction of future behaviour and the calculation of "what if" scenarios using simulation methods.

Next generation tools are needed to make the development, construction monitoring and analysis of such systems easier, faster, more reliable and - most importantly - understandable for decision makers and other stakeholders. The EUROSIM Technical Committee DDSS 'Data Driven System Simulation' [2] aims to support and coordinate combined research in the following areas:

Data. Integration, storage, management and analysis of very large data sets, unstructured data, secure and reproducible data management from sensors, IoT and different data sources such as dynamic databases or unstructured information sources. Development and Operation of Digital Twins and Synthetic Data Interfaces, Data acquisition, interfaces and analysis methods from statistics, machine learning and visual analysis.

Model. Formal, scalable modelling of different systems, heterogeneous modelling of subsystems and integration of these subsystems, development of modelling methods for computationally complex systems, multimethod modelling, including coupling and comparison based on data, system knowledge and application requirements. Development of innovative methods in numerical mathematics, co-simulation, hybrid simulation.

Processes. Linking data and models to simulation tools for complex systems and methods for reproducibility of results. Interfaces and visualisation of simulation results, decision support systems and the future development of Human-Computer Interaction (HCI).

Guidelines. Standardisation of the processes mentioned, modularisation of models, connectivity of simulations, comparability of the tools used as well as international guidelines on privacy, security, how to implement, test and quality assure specific technologies and how to integrate stakeholders.

In this article, the EUROSIM Technical Committee presents a first draft for ten concepts that should be considered for the implementation of modern and adequate simulation models.

2 Methods

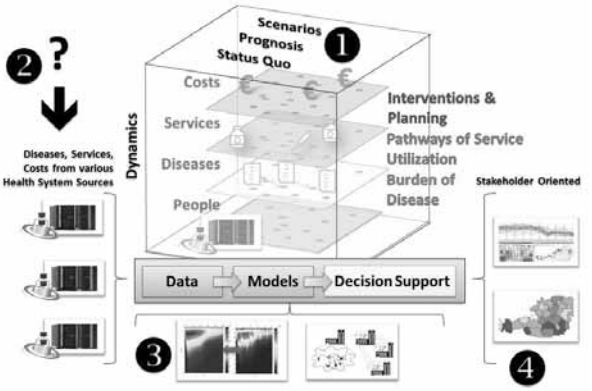


Figure 1: Schematic Overview DEXHELPP Infrastructure & Process (2014).

On basis of experiences of the Austrian DEXHELPP Competence Centre for Decision Support in Health Policy and Planning [3], which started in 2014 a concept was developed how large, interdisciplinary teams can handle these complex processes in the future and what are similarities and differences between health systems and other complex man-made systems. DEXHELPP developed an innovative research infrastructure with (1) a flexible virtualised health system, (2) methods to cope with data, (3) an adaptive analysis and simulation methods pool and (4) stakeholder oriented interfaces to enable researchers and other stakeholders to share data and methods for research and decision making (see Figure 1).

Within the framework of the development of this platform, corresponding methods were developed in six different sub-areas on the one hand, and on the other hand the methods were tested in practice with partners from the Austrian health system. Ten different areas in which there is a need for action were derived from this. These were compared with other disciplines within the framework of EUROSIM and expanded accordingly.

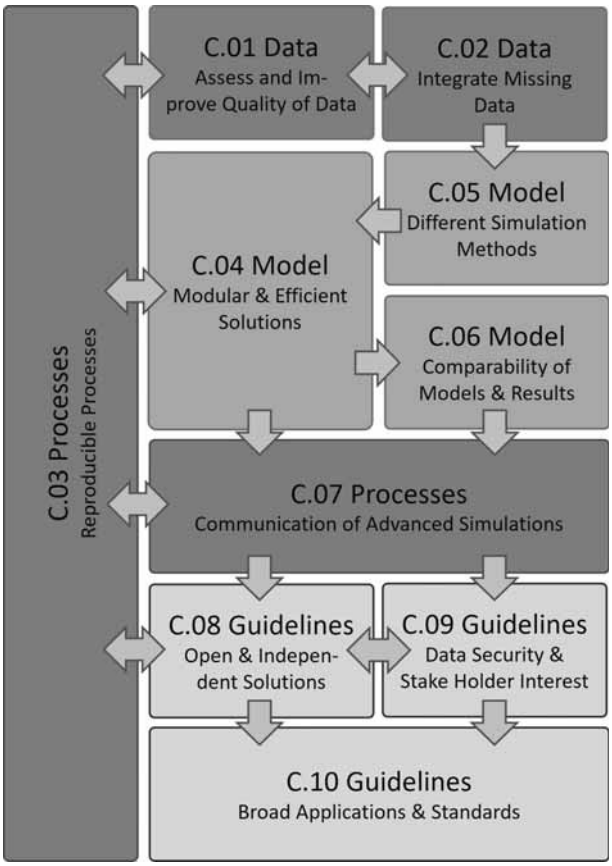


Figure 2: 10 Concepts to Integrate in Simulation Processes.

10 Concepts to Integrate were identified and presented at Invited Talks at University of Rostock [4] in 2018 and University of Stuttgart [5] in 2019 and are under development and extension with input from researchers in other applications and domains. The ten concepts are shown in Figure 2 and Table 1).

The concepts are structured into four thematic categories, which are **"Data"**, **"Model"**, **"Processes"** and **"Guidelines"**. This article does not include basic process mechanisms of modelling and simulation technology, which can be found in the literature. The concepts listed are state of the art and work in progress in their respective research areas and part of current research work. This paper aims to summarise their necessity and value for the development and operation of sustainable simulation studies.

3 Data

In this section challenges, ideas, examples, and benefits in the area of data integration into simulation studies are summarised.

	Concept
C.01	Assess and Improve Quality of Data
C.02	Integrate Missing Data
C.03	Reproducible Processes
C.04	Modular & Efficient Solutions
C.05	Different Simulation Methods
C.06	Comparability of Models & Results
C.07	Communication of Advanced Simulations
C.08	Open & Independent Solutions
C.09	Data Security & Stake Holder Interest
C.10	Broad Applications & Standards

Table 1: Overview Concepts.

Two concepts are included: **C.01 "Assess and Improve Quality of Data"** and **C.02 "Integrate Missing Data"** (shown as "Data" in Figure 2) offer the possibility to find wrong data and correct them, ideally also during the simulation process. For this purpose, methods of interactive visualisation and statistics are used, among others, to pre-process data collected unilaterally, e.g. sensor data or reimbursement data or to link data that are unstructured or have different structures without existing direct linkage. A particular challenge in simulation studies is the fact that not only the data itself can change during the project, but also the quality and structure. Furthermore, it may become necessary to integrate new data sources.

3.1 Challenge

The integration of large and heterogeneous data sets is an enormous challenge for classical simulation models. It is probably a central aspect why classical simulation has meanwhile fallen behind data science and machine learning approaches in some areas. In the areas mentioned, the integration of these data sources is in the foreground and is thus "thought of" from the beginning of a project. Classic simulation projects often focus on the development of methods and only later on data integration. Especially the lack of flexibility in data integration is often a challenge that is difficult to solve.

Concrete challenges in the area of data acquisition and data processing are the bias of collected data. Whether it is sensor data in technical applications or data collected in the case of the health system, e.g. in billing systems, this data must always be seen in the context of the purpose for which it was collected.

For example, in the health system this means that data collected for the purpose of billing often has a bias in that more expensive services are billed more often than they actually occur. In case of doubt, it is highly likely that, given two possible types of documentation, the one that is of greater benefit to the person, who documents will be chosen. However, if this data is used to estimate the burden of disease, the input parameters of the model are bound to be incorrect. In addition, pre-processing by different stakeholders (i.e. hospitals with different hospital information systems or doctors' practices) or the need to anonymise the data at an early stage of processing for data protection reasons pose further challenges. The handling of these points should be summarised and described in simulation projects in **C.01 "Assess and Improve Quality of Data"**. Furthermore, the data used in large simulation studies will usually not come from only one source. The question of how different data sets, which may be used, for example, in agent-based models for the parameterisation of the individual agents, must be brought together should be at the centre of considerations from the outset. Special attention should be paid to the fact that new data sources may be added in the course of projects lasting several years. An example of this is the development of a new therapy that is recorded in a new registry. This should be documented in a section on **C.02 "Integrate Missing Data"** and monitored continuously during the simulation study.

3.2 Implementation & Examples

For implementing **C.01**, a number of innovative methods are available. In DEXHELPP, the use of Explorative Visual Computing -Visual Analytics and Computational Statistics has proven to be an important building block in the implementation of the projects. The use of such methods makes it possible to identify outliers in data at an early stage in the simulation study, which have arisen through collection or preprocessing. A major limitation is that the reason for the bias is not directly apparent and the process view should be added to classical quantitative methods. In DEXHELPP, many projects were implemented with methods developed at VRVis (Centre for Virtual Reality and Visualisation). One example is [6]. In order to be able to evaluate a large number of time series with regard to their data quality, it is necessary on the one hand to automate the processes, but on the other hand to use necessary meta-information about the semantics of both the time series and the plausibility checks in order to structure and summarise the results of data quality checks in a flexible way. Already in this phase it is important to implement a comprehensive task analysis with domain experts and to derive processes from it in order to link quantitative methods with system knowledge. An example for statistical methods for the analysis of compositional data is [7]. Compositional data analysis refers to analysing relative information, based on ratios between the variables in a data set. In contrast to the interpretation as absolute information, it can be shown that already in data preprocessing not only different input parameters can be generated by means of univariate as well as multivariate statistical analyses, but that also important interpretations of these data for the modelling process arise.

Regarding the concept **C.02**, the necessary methods can be divided into two areas. On the one hand, suitable possibilities for the integration and linkage of (new) data sources must be embedded into the data processes, and on the other hand, parameterisation and calibration must be implemented with suitable methods for the modelling. Often, no unique identifiers are available for the record linkage of data. Agents can therefore not be parameterised appropriately. Therefore, the development of deterministic [8] and stochastic linkages is necessary [9]. Furthermore, the sustainable integration and combination of basic demographic data, structural socio-economic data and survey data is an example of a typical challenge in the health system. Optimisation and allocation algorithms are used in [10] to construct a structured population with corresponding temporal close-proximity interaction network from this data. This is particularly important for developing modular and reusable models. An example for linking such statistical population data with other data can also be found in [11]. Here, epidemiological data that are also used for the population model is combined with election data, e.g. voter turnout or arrival times of voters. Other necessary methods include the integration of AI methods to parameterise models based on historical data sets. In [12], for example, historical data of railway operations is used to parameterise a delay prediction model with a special focus on feature selection.

It is of particular interest that the two steps of integration and record linkage of data sources on the one hand and parameterisation and calibration on the other hand are in a direct interplay. I.e. the integration of new available data must be possible, but this directly results in new necessities as to how the model is parameterised

and calibrated. Conversely, when the simulation is implemented with the domain experts, knowledge will continuously be gathered that leads to the necessity of integrating new data sources. Even if the problem is, of course, well known and described in principle, special attention must be paid to efficiency (automation) and feasibility with large and heterogeneous data sources as well as complex models during method development.

3.3 Benefit

The briefly mentioned methods are examples of which tasks in this area must not only be implemented, but also documented in a reproducible and comprehensible way in order to be prepared for the next points. A concrete goal (and evaluation criterion) is the possibility to identify wrong data sets and to exclude them from the further modelling process. With the potential to identify possible causes for the errors, a first benefit of the modelling process would be added. Furthermore, it must be possible to change wrong data sets, even over the time of the simulation project and after the data sets have been integrated and to include new data sets and data structures. We can process data to use it in subsequent simulation studies. This means that suitable methods are available to parameterise, calibrate and validate the model.

4 Model

To develop **C.04 "Modular & Efficient Solutions"** using **C.05 "Different Simulation Methods"** and maintain **C.06 "Comparability of Models & Results"** (summarised as "Model" in Figure 2) includes sustainable, modular models that can be quickly adapted to new problems and concepts for comparing, combining and linking models (qualitatively and quantitatively). The basic idea is that there is not only one methodology for modelling. The process of which method was chosen should be clearly presented, and the possibility of comparing or coupling models should be discussed if applicable and usable. Qualitative and quantitative comparison to analyse limitations of modelling approaches and implementation is possible, as well as methods like parameter transformation between models. The models should be modular in the case of usability in other areas so that, starting from data integration, the modules can be reused with minimal effort.

4.1 Challenge

Often, modelling methods proposed in the literature for dealing with the given questions are used for simulation projects - in line with good scientific practice. However, due to the change in available data described in the chapter "Data", emerging system knowledge and adopted research questions [13] there is often the possibility to try other modelling approaches. This potential should be used, but there is a risk of getting bogged down in the task: We need transparent, "simple" models.

In this respect, concepts should be developed to be able to implement modular simulation parts. Their reusability also serves to increase quality and sustainability. This is also necessary and helpful insofar as in many cases no established simulators can be used in practice for runtime reasons, but the solutions are programmed out fundamentally after a rapid prototyping and proof of concept phase. Therefore, the solutions should be kept as complicated but also as simple as possible. Especially in the case of complex processes, such as the use of buildings over time, it will make sense to couple existing models, i.e. not to develop new solutions that have already been tested and validated. However, this poses the challenge that (at least) interfaces and runtime behaviour have to be validated and documented again. These issues should be reviewed and addressed in item **C.04 "Modular & Efficient Solutions"**.

Based on this point, it becomes clear that there is a need to implement different levels of detail and different issues with different approaches. Therefore, it is necessary to define clear processes according to which criteria a model concept was selected for implementation. In this respect, simulation theory is now intensively concerned with the question of how models can be compared at all and how the "right" model can be selected: Based on this development, corresponding steps in the modelling process should be documented. An example would be the sensible representation of an epidemic spread by agents, if concrete interventions such as school closures are to be simulated and these cannot be represented in a comparable differential equation approach. Conversely, for the analysis of a basic system behaviour, a differential equation model (and the existing methods for analysis) should be used whenever possible. A possible combination of model parts for different (sub)systems should be considered and checked. The motivation for the choice and limitation of the cho-

sen method, which arises from the data situation, the research question or the system knowledge, should be presented clearly and evidence-based in **C.05 "Different Methods for Different Research Questions"**. Under no circumstances should personal preferences of the modeller play a role here.

Last but not least, **C.06 "Comparability of Models & Results"** is by no means only about the obvious Comparability of Models & Results resulting from C.05, in the case that a question was calculated with two different models, but also about the possibility of making the model processes themselves comparable. This raises questions such as how parameter sets can be exchanged between microscopic and macroscopic models. Furthermore, it should be possible to better work out the limitations of individual approaches by comparing models. This will not be possible in every simulation study, but should be considered as a fundamental possibility.

4.2 Implementation & Examples

An approach for sustainable use of individual model parts, as should be fulfilled in **C.04**, is the development of a generic population model within the framework of DEXHELPP. The Generic Population Concept (GEPOC) has been developed since 2014 [14] and makes it possible to map different countries, flexibly integrate different sets of input parameters and use different modelling techniques (Agent Based, Discrete Modelling, System Dynamics). The population model is an example of the usefulness of modularisation, as the model cannot only be used in one application area, but the effects of interventions on the population play an important role in many areas. Examples of this are, in addition to the intervention and supply analysis in the health sector, also the use in the modelling of new mobility concepts or in the area of energy supply.

The current implementation for Austria simulates the population of Austria between 1998 and 2100 in such a way that historical data match the data of Statistics Austria, but also the forecasts match the respective assumptions of the national statistical authority up to a defined (small) error. The standard model is basically without interaction, but capable of it - i.e. this aspect is also optional for reasons of efficiency. In microscopic modelling, for example, agent properties are date of birth, sex and place of residence (latitude/longitude). Here, the link to the chapter "Data" is also established, since the possibility of being able to parameterise an existing, modular model again and again with different data is of great advantage here. The model was used, among other things, to advise the Austrian federal government during the Covid-19 crisis [15] and has proven itself to enable fast, flexible and quality-assured model implementation.

Different model derivations are managed on Git Hub as individual branches. Modules can be added step by step. In the case of Covid-19 modelling, these are a variety of different aspects, such as exact place of residence, or immunisation status. A fundamental example is the contact module, which is important for modelling dispersal. This was developed as part of an influenza modelling project starting in 2010 and implements the contact networks based on the POLYMOD study, a large survey of infection-related contact patterns, on the characteristics of 97,904 contacts recorded with 7,290 participants. Two aspects should be emphasised here: on the one hand, it is necessary to appropriately extrapolate the data-based contacts using statistical methods. On the other hand - based on model assumptions, as in the case of Covid-19 through contact restrictions, lock-downs and other measures - the modelled contact networks change. This must therefore be possible and as efficient as possible in the implementation. Accordingly, the model can also be used for effects in other areas, up to the analysis of possible couplings with other modelling approaches in multimethod modelling [16] or development of new interdisciplinary approaches [17].

Dealing with different models **(C.05)** was a starting point for the DEXHELPP platform. A rather simple example of comparing ODEs, PDEs, difference equations and CAs [18] was extended over the years by agent-based models and the respective modelling process parts such as parameterisation and cross model validation.

The aforementioned GEPOC model can also be used to illustrate how a model comparison **(C.06)** can be carried out [19]. Thereby, the methodological possibility of comparison forms the basis to make competent decisions why certain decision support should be implemented with concrete methods [20] or to what extent models can be combined to hybrid approaches [21]. In the course of the Covid-19 crisis, model comparisons were used in Austria in advising the federal government by implementing three different model approaches and comparing them on a weekly basis [23] , a current example of the comparison of models used

(also internationally) is the ECDC Covid-19 Scenario Hub [24] The approaches mentioned are not limited to the level of population modelling, an example of which is [22]. The microscopic behavioural aspects like motion and proliferation of 'pigment cells'of the human skin are implemented using basic principles of agent-based simulation whereas the complex geometry of the microphysiological environment of melanocytes is modeled using the techniques of differential geometry. The combination of a small-scale behavioural model and the interaction with the complex environment allows to simulate and reproduce the growth of melanocytic skin lesions in silico.

4.3 Benefit

The aspects of modelling mentioned in the section are fundamentally linked to the data processes described in the section before. Assuming that a stable and validated modelling has been implemented, we can assume that with reasonable effort - if the requirements are not adapted - a change in the model strategy and a change in the parameterisation will not lead to any significant advantage. In contrary with changing requirements the model can be flexible improved if necessary. The possibility to represent the heterogeneity of the system under consideration is sufficiently given and the level of detail of the modelling is justified. This is supported by the possibility (and ideally implementation) of quantitative as well as qualitative comparison of different, methodically cleanly comparable methods. Differences in the results of different models are reasonable - as different model assumptions, aggregations or focus are set - and explained. Parameters can be transferred between diverse implementations and models. They can either be combined through multi-method modelling or suitably coupled through co-simulation if required. The implemented processes can clearly show the limitations of modelling and implementation.

5 Processes

C.03 "Reproducible Processes" and **C.07 "Communication of Advanced Simulations"** (Blocks 3 and 7 in Figure 2) are essential to guarantee the credibility and usability of the models and are decisive for the impact of decision-support models. Tools to manage and share data (e.g. [30] and concepts to communicate not only the simulation results, but also the modelling process and model construction are used. The reproducibility of processes and Data Citation Principles applied on all data sources is indispensable to be able to repeat the simulation studies at any time and thus increase the credibility of simulation technology itself in the medium term. Modelling steps such as the implementation of stochasticity, coupling of model parts and others are clearly documented. Selection and presentation of the results is justified, the conclusions are clearly presented and are related to the simulation results, especially when outcomes are relative (e.g. prioritisation of interventions) or qualitative. Improving the comprehensibility of the modelling process, simulation use and results of different categories through Data Representation and Human Computer Interfaces (HCI) and other strategies is essential to achieve the real purpose of simulation in the field of decision support, namely sound change management.

5.1 Challenge

From modelling practice we know that the data landscape is usually heterogeneous and that the data situation changes continuously in the modelling process and during the use of simulations. Different models with regard to different time scales, granularity of the representation of system variables and outputs as well as different properties make it difficult to keep simulation studies as reproducible as possible. In addition, the necessary and important regulations and standards regarding data security, personal protection and confidentiality must be observed. Nevertheless, in order to achieve credibility, it is essential that simulation studies in the future - like real experiments - should be reproducible worldwide in the laboratory, starting with the documentation of input data and parameters, through the systemic and strategic assumptions, the modular model parts and their documentation, to the specification of stochastic process assumptions. Corresponding methods must be developed and used in **C.03 "Reproducible Processes"**. Furthermore, projects still fail to reduce the model and simulation complexity in the direction of the decision-makers. On the one hand, complicated models are built precisely in order to do justice to the heterogeneity and dynamics of modern sociotechnical processes, the reduction of which may not be (sensibly) possible. On the other hand, these models are not acceptable to the decision-makers because they are not comprehensible. In this respect, possibilities must be created (or used) in **C.07 "Communication of Ad-**

vanced Simulations", which can present models in a suitable way, can clearly present relevant mechanisms of action and break down the results of dynamic processes in a suitable way. Need for Change Management & Interdisciplinarity.

5.2 Implementation & Examples

As described in chapter "Data", one challenge is to deal with the changing data situation. New data is added, which not only fills existing structures with new data, but also changes the structures themselves **(C.03)**. Specific subsets of data are selected that fit the research question at hand. In this respect, we need methods to identify versions of a subset. Recommendations for this were developed, for example, by the RDA Working Group on Dynamic Data Citation (WGDC) in [25] with versioned data, timestamps and a query-based mechanism for subset formation. We also need fair data use, including concepts for managing the life cycle of research data that can be machine-processed with Data Management Plans (maDMPs), as in [26]. Simulation models can also be used to fill in missing data or generate new data. Data farming [27] enables the use of simulation models to generate data that can also be used for other methods such as machine learning. Simulation thus serves as a complement to observational data, the connection of which requires innovative methods to couple simulation, real-time data sources and the traditional historical data to verify and validate models [28]. But the convergence of physical and virtual worlds now goes far beyond this in cyber-physical systems (CPS), of course. The digital twin represents the maximum challenge, serving as a digital image of the physical world from the planning phase through strategic planning to operational use. It consists of a set of adaptive models that mimic the behaviour of a physical system and are updated along its life cycle by real-time data [29]. The digital twin has both simulation capabilities that can approximate the behaviour of the real system and emulation capabilities that allow the digital twin to synchronise with the real system and thus duplicate and mimic the physical system in the real world. The digital twin therefore offers more accurate replication compared to the simulation model and represents a new paradigm in modelling and simulation [29].

Last but not least, this also involves reproducibility at the "other end" of the simulation process, namely the use of the results of large agent-based models as a synthetic data source, as for example presented in the Covid-19 crisis here [30]. Other aspects include the conflicting goals of protecting and controlling sensitive data on the one hand, while allowing third party access on the other, as described in [31]. This is a major challenge especially for simulation studies, as the choice of modelling method is also affected here. The better the mechanisms are implemented in the data selection, the more detailed models can be applied. Here, the close connection of data, i.e. parameterisation, calibration and validation, to the model structure becomes apparent. Specifically for agent-based modelling in archaeology, this is outlined in [32], where the issue is up to clearly defining what stochastic range needs to be achieved in corresponding ones (see [33]) and how this can be mapped in the reproducibility discussion.

On the way to credibility and usability, achieving (and communicating) reproducibility is only one pillar. In addition, the communicability of the model results, the modelling process and the simulation study itself is equally decisive **(C.07)**. The importance of this was already shown in 2017 in [34] by means of an analysis of the extent to which people are more willing to be vaccinated if the benefits of vaccination are presented to them more clearly. This is a crucial aspect for the fruitful use of simulation models, as recently became apparent in the Covid-19 crisis. Even in phases in which the benefits of certain interventions were quite provable (in other phases these benefits were quite controversial), it became increasingly difficult to communicate these benefits widely. Three aspects are currently being researched intensively in this context: First, corresponding models must be clearly documented and communicated. Black box models whose structure is not comprehensible or understandable will justifiably not generate any benefit in the future, be it in the analysis of climate change or all other questions. Secondly, their use must be clearly documented and transparently implemented. In the field of health systems research, there are established processes for how questions are defined, how they are processed and how they are finally evaluated. In the case of the vaccine evaluation of Covid-19 in Austria, this was implemented and published in [35] including the involvement of a steering board. Last but not least, it is about clear communication of the results, as they are researched in the visualisation community. Here, not only are clear concepts for evaluation implemented, but these are evaluated themselves, as in [36]. Simulation research is still some steps away from this status.

5.3 Benefit

When implementing the above concepts, the results is not only a well-parameterised model with reasonable data sources, but also a simulation study that finds acceptance in the respective field of application. This can never be completely described technically, but two points are covered as well as possible: Firstly, the credibility of the simulation study was implemented with suitable documentation of the data used, the selected model modules, the implementation of the studies and all other framework conditions in such a way that the respective state of the art of reproducibility valid at the time of implementation was achieved. The experiment is reproducible. Secondly, the study can be appropriately communicated to those experts and the general public in accordance with the current state of the art. This also applies to the model structure, the implementation of the simulation study and the results. The study is comprehensible. Once this has been done to a sufficient extent, there is still the question of the connection to international standards, how a balance can be found between the needs of the clients (who also provide the data) and transparency and open access, as well as the question of domain-specific and interdisciplinary standards. These will be described in the next section.

6 Guidelines

Last but not least, guidelines, standards that go beyond the concrete implementation are crucial (Blocks 8-10 in Figure 2). Here, the concepts of **C.08 “Open & Independent Solutions”**, **C.09 “Data Security & Stake Holder Interest”** and **C.10 “Broad Applications & Standards”** are crucial. The possibility of publication is limited, for example, by (justified) economic or data protection interests, which, however, leads to a lack of comparability of different models and thus jeopardises quality. This requires fundamental regulations such as those addressed in the General Data Protection Regulation and Data Governance Act. Clear and transparent processes are necessary for every project (even before the start of a simulation development) as well as the reuse of models is necessary to ensure quality and sustainability over time. Standards for different domains have to be established and - as a vision - should be valid for simulation studies in all domains.

6.1 Challenge

The basic "possibility" of achieving credibility (as described in the previous chapter through reproducibility and comprehensibility) is currently often limited in reality for "external reasons", i.e. not due to the technical implementation of the simulation study. Data, model structure and documentation of the simulation study are often not published due to data protection, interests of data owners or clients. At the same time, a lack of comparability of (published) models, simulations and results leads to a lack of credibility of the discipline itself. Exaggeratedly formulated, one could write that in the case of simulation research, the lack of transparency leads to the discrediting of the entire discipline, as this lack is not attributed to a concrete implementation, but to the concept of simulation. Furthermore, non-publication prevents the further improvement of the quality of standardised model modules. For this reason, **C.08 "Open & Independent Solutions"** is an even more important concept and rules and guidelines regarding this will play an important role. This leads directly to the challenge of how to (pre-)define and guarantee stakeholder interests. This is the only way to clarify justified (or unjustified) objections at an early stage and to establish clear guidelines on the extent and aggregation of results that may be published. This is also the motivation for the concept **C.09 "Priority for Data Security and Stakeholder Interests"**, because only through this prioritisation will the necessary willingness to receive data and to be able to implement the publications to a sufficient extent be achieved. Of course, legal frameworks are still necessary and important. On the one hand, these must keep data protection in mind (GDPR) and, on the other hand, enable the necessary publication in the "public interest" in order to enable transparency in decision-making processes and efficient control of systems and processes. Through publication and the associated possibility of reusing models interdisciplinarily, on the one hand resources for the new and further development of models can be better used worldwide, and on the other hand the quality of the models can be better and more sustainably ensured. Examples of this would be the establishment of population models that follow the same standards worldwide and on which interventions can be simulated. This is an urgent challenge because - taking climate change and successful possible interventions as an example - domain boundaries are already history and individual aspects can no longer be considered singularly. Cur-

rently, the development of standards in sub-domains is at different stages of progress. Interdisciplinary standards are an urgent challenge that should be solved as soon as possible. Solutions should be developed in **C.10 "Broad Applications & Standards"** and implemented or referenced in individual simulation studies.

6.2 Implementation & Examples

The consequences of non-transparent, non-reproducible processes are twofold, for example in the case of concrete decision support in the area of the health system. Lack of credibility of results on the one hand and lack of availability of data or reproducible model assumptions (in a more general sense) on the other. The second aspect seems to be more cause than effect, but it turns out as follows: the knowledge that results of calculations are not shared and further used in a quality-assured way leads to irreconcilable differences between stakeholders as to who should provide which data and how they are processed. Especially in systems like the Austrian health system, where resources and decisions are shared between several stakeholders (in the case of the Austrian health system government, federal states and social insurance), lack of process quality are good arguments for not participating in a common analysis strategy. The result is diffuse or contradictory bases for decision-making.

Concepts **C.08** and **C.09** should therefore be considered together. On the one hand, open and transparent processes are needed; ideally, both the data sources and the processes and results should be published **(C.08)**. Open access journals, corresponding data platforms (described in the previous chapters) and GitHub servers are suitable for publishing the source code. Corresponding access points have been and are being continuously developed in the respective communities and should be sustainably linked to activities of EUROSIM and other simulation societies. In the context of concrete political decision-making processes, one should go one step further. In the analysis of the current immunisation against Sars-CoV-2, an up-to-date, model-based evaluation of this immunisation was published monthly within the framework of DEXHELPP [38], as concrete discussion processes were also continuously accompanied on the basis of this assessment.

On the other hand, it must already be clear before the start of a simulation project which partners provide which data under which boundary conditions **(C.09)**. In the context of governance, it must be clarified which stakeholders and scientific partners have access to which data aggregates, how and where these are linked and processed with which methods, and which results are published in which resolution by which stakeholder. These aspects must be clarified legally, technologically and formally and documented in writing. The two aspects C.08 and C.09 are directly related and it is short-sighted to think that the data issue can be considered in isolation from the methodology. A proof of concept for the whole process was implemented as an integrated solution, the DEXHELPP Research Server, within the COMET project DEXHELPP [37].

Last but not least, modularisation for the purpose of a cross-domain use of reusable, quality-assured models as well as guidelines and the standardisation of methods and their use are important steps for the future use of simulation in decision support **(C.10)**. In health systems research, the SMDM/ISPOR Modeling Good Research Practices [39] should be mentioned as an example, which define quality .and selection criteria for methods. It will also become necessary to define these standards and guidelines across domains. Just as energy consumption, energy transfer, energy storage and other aspects must be considered together, this also applies to simulation studies and models of the future. From the health system to mobility and climate, integrated models will be needed in the future in order to be able to depict the heterogeneity and dynamics appropriately.

6.3 Benefit

In the final chapter, the areas of open access and public domain were briefly outlined, especially in connection with the protection and planning of stakeholder interests, as well as the question of modularisation, standardisation and guidelines. In a simulation study, clear rules are documented in advance as to which results are published in which aggregation and which are not. The non-publication is justified, as are the legitimate interests of all parties involved. Methods are available to reuse as many model parts as possible under regulated framework conditions, either on one's own or in the research network.

7 Summary

There are many developments in the respective communities. At this point, an attempt has been made to briefly outline - in a first statement - which aspects are

specifically crucial for modern simulation studies and which points are recommended for attention for specific projects. The aim is to show a spectrum of possibilities that should be considered, tested and ideally integrated as far as possible.

Some of the major challenges (and weaknesses) are exemplary: Centralised data bunkers, are too inflexible to keep up with ongoing changes. Models that are not scalable according to data, new system knowledge or research focus - and that are not comparable - weaken credibility and usability. Lack of interfaces and willingness to cooperate between models and other methods prevent optimal solutions.

We therefore need **Flexibility**, through decentralised (and secure) storage and documented, professional processes. **Sustainability** through modular models and simulations as well as **Appreciation** between methods from data science, mathematics and simulation.

References

[1] Covid-19 Simulation in Austria, http://www.dexhelpp.at/en/appliedprojects/tmp/covid-19/, [Online; accessed 8.9.2022]

[2] EUROSIM Technical Committee "Data Driven System Simulation", https://www.eurosim.info/tcs/tc-ddss/, [Online; accessed 8.9.2022]

[3] DEXHELPP, http://www.dexhelpp.at/en/project-description/state-of-the-art/, [Online; accessed 8.9.2022]

[4] Abstract Colloquium Talk 14.6.2018 "Sharing Data, Methods, a. Simulation Models - New Opportunities for Digital Health Care", https://www.informatik.uni-rostock.de/veranstaltungen/detailansicht-des-events/n/kolloqiumsvortrag-von-nikolas-popper-director-dexhelpp-wien-coordinator-centre-for-computational-complex-systems-tu-wien-46337/ [Online; accessed 8.9.2022]

[5] Abstract Faculty Talk 25.3.2019 "Integrated Processes for Modelling & Simulation", https://www.sfbtrr161.de/newsandpress/events_sfbtrr161/pastevents/, https://www.sfbtrr161.de/newsand press /downloads/Stuttgart_Faculty_Talk _20190325 _PrintVersion_s.pdf [Online; accessed 8.9.2022]

[6] Arbesser C, Spechtenhauser F, et al. Visplause: Visual Data Quality Assessment of Many Time Series Using Plausibility Checks. *IEEE Transactions on Visualization and Computer Graphics* 23, Nr. 1 (Januar 2017): 641–50. DOI 10.1109/TVCG.2016.2598592.

[7] Mert MC, Filzmoser P. Endel G, Wilbacher I. Compositional Data Analysis in Epidemiology. *Statistical Methods in Medical Research* October 6, 2016, 096228021667153. doi:10.1177/0962280216671536.

[8] Popper N, Glock B, et al. Deterministic Record Linkage of Health Data as Preparatory Work in Modelling and Simulation-Use Case: Hospitalizations in Austria. *Proceedings of the 6th International Workshop on Innovative Simulation for Health Care*. pp. 44-49 (2017).

[9] Glock B, Endel,F. et al. Challenges and Results with the Record Linkage of Austrian Health Insurance Data of Different Sources. *Informatics for Health Conference* 2017 (24 – 26. April, Manchester, UK)

[10] Schneckenreither G, Popper N. Dynamic multiplex social network models on multiple time scales for simulating contact formation and patterns in epidemic spread. *2017 Winter Simulation Conference (WSC)* 2017, pp. 4324-4336. DOI 10.1109/WSC.2017.8248138

[11] Weibrecht N, Roessler M, et al. How an election can be safely planned and conducted during a pandemic: Decision support based on a discrete event model. *Plos one* 2021, 16(12).

[12] Leser D, Wastian M, et al. Comparison of Prediction Models for Delays of Freight Trains by Using Data Mining and Machine Learning Methods. *SNE Simulation Notes Europe* 2019, 29(1), 45–47. DOI 10.11128/sne.29.sn.10467

[13] Popper N. Comparative modelling and simulation a concept for modular modelling and hybrid simulation of complex systems *Phd Thesis* 2015, TU Wien Repos. DOI 10.34726/hss.2015.21448

[14] Bicher M, Urach C, Popper, N. GEPOC ABM: A generic agent-based population model for Austria *in Proceedings of the 2018 Winter Simulation Conference* 2018, pp. 2656-2667. DOI 10.1109/WSC.2018.8632170

[15] Bicher M, Rippinger C, Urach C, Brunmeir D, Siebert U, Popper N. Evaluation of Contact-Tracing Policies against the Spread of SARS-CoV-2 in Austria: An Agent-Based Simulation. *Medical Decision Making, 41-8*. pp. 1017-1032 (2021).

[16] Glock B, Popper N, Breitenecker F. Various Aspects of Multi-Method Modelling and its Applications in Modelling Large Infrastructure Systems like Airports. *The 27th European Modeling and Simulation Symposium* 2015, (pp. 197-206). Proceedings of the European Modelling and Simulation Symposium 2015.

[17] Heinzl B, Rößler M, Popper N, et al. Interdisciplinary strategies for simulation-based optimization of energy efficiency in production facilities. *UKSim 13th International Conference on Computer Modelling and Simulation* 2013, pp. 304-309, IEEE.

[18] Schneckenreither G, Popper N, Zauner G, Breitenecker F. Modelling SIR-type epidemics by ODEs, PDEs, difference equations and cellular automata – A comparative study. *Simulation Modelling Practice and Theory* 2008, 16(8), 1014-1023.

[19] Bicher M, Glock B, Miksch F, Popper N, Schneckenreither G. Definition, validation and compari-son of two population models for Austria. *Int. Journal of Business and Technology* 2015, 4(1), 7.

[20] Miksch F, Jahn B, Espinosa KJ, Chhatwal J, Siebert U, Popper N. Why should we apply ABM for decision analysis for infectious diseases? An example for dengue interventions. *PloS one* 2019, 14(8).

[21] Emrich S, Breitenecker F, Zauner G, Popper N. Simulation of influenza epidemics with a hybrid model-combining cellular automata and agent based features. *ITI 30th International Conf. on Information Technology Interfaces* 2008, (pp. 709-714). IEEE.

[22] Schneckenreither G, Tschandl P, Rippinger C, et al. Reproduction of patterns in melanocytic proliferations by agent-based simulation and geometric modeling. *PLOS Computational Biology* 2021, 17(2): e1008660. DOI 10.1371/journal.pcbi.1008660

[23] Bicher M, Zuba M, Rainer L, et al. *Supporting Austria through the COVID-19 epidemics with a forecast-based early warning system.* 2020, medRxiv, 2020-10. DOI 10.1101/2020.10.18.20214767

[24] ECDC Covid-19 Scenario Hub, https://covid19scenariohub.eu/ [Online; accessed 4.11.2022]

[25] Rauber A, Asmi A, Van Uytvanck D, Proell S. Identification of reproducible subsets for data citation, sharing and re-use. *Bulletin of IEEE Technical Committee on Digital Libraries* 2016, Special Issue on Data Citation, 12(1), 6-15.

[26] Blumesberger S, Gänsdorfer N, Ganguly R, et al. FAIR Data Austria – Abstimmung der Implementierung von FAIR Tools und Services. *Mitteilungen Der Vereinigung Österreichischer Bibliothekarinnen und Bibliothekare* 2021, 74(2), 102–120. DOI 10.31263/voebm.v74i2.6379

[27] Sanchez SM. Data farming: Methods for the present, opportunities for the future. *ACM Transactions on Modeling and Computer Simulation (TOMACS)* 2020, 30(4), 1-30. https://doi.org/10.1145/3425398

[28] Scala P, Mota M, Blasco-Puyuelo J, Garcia-Cantu O, Blasco C. A novel validation approach for validating the simulation model of a passengers' airport terminal: case study Palma de Mallorca airport. *EMSS 2022* 2022, Rome, Italy.

[29] Semeraro C, Lezoche M, Panetto H, Dassisti M. Digital twin paradigm: A systematic literature review. *Computers in Industry* 2021, 130, 103469.

[30] Popper N, Zechmeister M, BrunmeirD, et al. Synthetic Reproduction and Augmentation of COVID-19 Case Reporting Data by Agent-Based Simulation. *Data Science Journal, 20(1)*, pp. 16ff (2021).

[31] Weise M, Kovacevic F, Popper N, Rauber A. OSS-DIP: Open Source Secure Data Infrastructure and Pro-esses Supporting Data Visiting *Data Science Journal,* 2022, 21(1).

[32] Popper N, Pichler P. *Reproducibility. In Agent-based Modeling and Simulation in Archaeology* 2015, pp. 98, Springer, Cham.

[33] Bicher M, Wastian M, Brunmeir D, Popper N. *Review on Monte Carlo Simulation Stopping Rules: How Many Samples Are Really Enough?* 2022, Simulation Notes Europe, 32(1), 1-8. DOI 10.11128/sne.32.on.10591

[34] Betsch C, Böhm R, Korn L, Holtmann C. *On the benefits of explaining herd immunity in vaccine advocacy.* 2017, Nature human behaviour, 1(3), 1-6.

[35] Jahn B, Sroczynski G, Bicher M, Rippinger C, Mühlberger N, Santamaria J, et al. *Targeted covid-19 vaccination (tav-covid) considering limited vaccination capacities—an agent-based modeling evaluation.* 2021, Vaccines, 9(5), 434.

[36] Isenberg T, Isenberg P, Chen J, Sedlmair M, Möller T. *A systematic review on the practice of evaluating visualization.* 2013, IEEE Transactions on Visualization and Computer Graphics, 19(12), 2818-2827.

[37] Popper N, Endel F, Mayer R, Bicher M, Glock B. Planning Future Health: Developing Big Data and System Modelling Pipelines for Health System Research. *Simulation Notes Europe* 2017, 27(4), 203-208. DOI 10.11128/sne.27.tn.10396

[38] Bicher M, Rippinger C, Schneckenreither G, Weibrecht N, Urach C, Zechmeister M, et al. *Model based estimation of the SARS-CoV-2 immunization level in Austria and consequences for herd immunity effects.* 2022, Scientific Reports, 12(1), 1-15.

[39] Caro JJ, Briggs AH, Siebert U, Kuntz KM *Modeling good research practices — overview: a report of the ISPOR-SMDM Modeling Good Research Practices Task Force.* 2012, Medical Decision Making, 32(5), 667-677.

SNE Simulation News

EUROSIM Data and Quick Info

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

ASIM SPL 2023
20th ASIM Dedicated Conference on Simulation in Production and Logistics
September 13-15, 2023, Ilmenau, Germany www.asim-fachtagung-spl.de

ASIM STS, GMMS & EDU Workshop 2023
March 6-7, 2023, Magdeburg, Germany www.asim-gi.org/magdeburg2023

EUROSIM CONGRESS 2023
July 3-5, 2023, Amsterdam, The Netherlands www.eurosim2023.eu

Contents

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

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), *m.mujica.mota@hva.nl*
Past President	Emilio Jiménez (CAE-SMSG), *emilio.jimenez@unirioja.es*
Secretary	P. M. Scala, *p.m.scala@hva.nl*
Treasurer	Felix Breitenecker (ASIM) *felix.breitenecker@tuwien.ac.at*
Webmaster	Irmgard Husinsky, *irmgard.husinsky@tuwien.ac.at*
SNE Editor	F. Breitenecker, *eic@sne-journal.org*

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 Editor-in-Chief	Felix Breitenecker *eic@sne-journal.org*

→ *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;

Due to Covid-19 virus some **EUROSIM** events had to be cancelled in 2020 or 2021, resp. 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***

EUROSIM Congress 2023, the *11th EUROSIM Congress*, will be organised by **DBSS**, the Dutch Benelux simulation society, in Amsterdam, July 3-5, 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 SPL** Int. Conference 'Simulation in Production and Logistics' – biannual
  ASIM SPL 2023 20th ASIM SPL, Sept. 13-15, 2023, Ilmenau, Germany ***www.asim-fachtagung-spl.de***
- **ASIM SST** 'Symposium Simulation Technique' – biannual
- **MATHMOD** Int. Vienna Conference on Mathmatical Modelling – triennial

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

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

CEA-SMSG – Spanish Modelling and Simulation Group

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

→ *http://www.ceautomatica.es/grupos/*

→ *emilio.jimenez@unirioja.es*
simulacion@cea-ifac.es

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

CEA - SMSG Officers	
President	José L. Pitarch, *jlpitarch@isa.upv.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 May 2022

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 nam. 13, 121 35 Praha 2, Czech Republic

CSSS Officers	
President	Miroslav Šnorek, *snorek@fel.cvut.cz*
Vice president	Mikuláš Alexík, *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 Alexík, *alexik@frtk.fri.utc.sk*
Web EUROSIM	Petr Peringer, peringer@fit.vutbr.cz

Last data update December 2012

DBSS – Dutch Benelux Simulation Society

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

→ *www.DutchBSS.org*

a.w.heemink@its.tudelft.nl

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

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

Last data update June 2016

LIOPHANT Simulation

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

→ *www.liophant.org*

info@liophant.org

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

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

Last data update June 2016

LSS – Latvian Simulation Society

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

→ *www.itl.rtu.lv/imb/*

Egils.Ginters@rtu.lv

Prof. Egils Ginters, Kirshu Str.13A, Cesis LV-4101,
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 November 2020

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 Durrhes, 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

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@iias.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@iias.spb.su*
Chair Man. Board	A. Plotnikov, *plotnikov@sstc.spb.ru*
Secretary	M. Dolmatov, *dolmatov@simulation.su*
Repr. EUROSIM	R.M. Yusupov, *yusupov@iias.spb.su* Y. Senichenkov, *senyb@dcn.icc.spbstu.ru*
Deputy	B. Sokolov, *sokol@iias.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	Tadeusz Nowicki, *Tadeusz.Nowicki@wat.edu.pl*
Vice president	Leon Bobrowski, *leon@ibib.waw.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 May 2022

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	Tiina Komulainen, *tiina.komulainen@oslomet.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 May 2022

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 KingdomUKSIM Officers

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

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.

→ *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, *florin_h2004@yahoo.com* Marius Radulescu, *mradulescu.csmro@yahoo.com*
Repr. EUROSIM	Marius Radulescu
Deputy	Florin Hartescu
Edit. Board SNE	Constanta Zoe Radulescu, *zoe@ici.ro*
Web EUROSIM	Florin Hartescu

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, *kozeta.sevrani@unitir.edu.al*
Repr. EUROSIM	Kozeta Sevrani
Edit. Board SNE	Albana Gorishti, *albana.gorishti@unitir.edu.al* Majlinda Godolja, *majlinda.godolja@feut.edu.al*

Last data update July 2019

Societies in Re-organisation / Former Societies

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

- CROSSIM – *Croatian Society for Simulation Modelling*
  Contact: Tarzan Legović, *Tarzan.Legovic@irb.hr*
- FRANCOSIM – Société Francophone de Simulation
- HSS – Hungarian Simulation Society
- ISCS – Italian Society for Computer Simulation

The following societies have been formally terminated:

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

HSS – Hungarian Simulation Society

There are plans to reactivate Hungarian Simulation Society. M. Mujica Mota EUROSIM President, is in contact with Andrási Gábor, Head of the Dean's office at the Faculty of International Management and Business of Budapest Business School University of Applied Sciences (BBS). We ask interested people to contact Mr. Gábor, *andrasi.gabor@uni-bge.hu*.

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
- Support of the Conference Series **MATHMOD Vienna** (triennial, in co-operation with **EUROSIM**, **ASIM**, and **TU Wien**) – www.mathmod.at
- Administration of **ASIM** (German Simulation Society) and administrative support for **EUROSIM** *www.eurosim.info*
- Simulation activities 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

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:

→ *www.sne-journal.org,*

ASIM Books – ASIM Book Series – ASIM Buchreihen

Monographs / Proceedings

Proceedings Langbeiträge ASIM SST 2022 -26. ASIM Symposium Simulationstechnik, TU Wien, Juli 2022
F. Breitenecker, C. Deatcu, U. Durak, A. Körner, T. Pawletta (Hrsg.), ARGESIM Report 20; ASIM Mitteilung AM 180
ISBN ebook 978-3-901608-97-1, DOI 10.11128/arep.20, ARGESIM Verlag Wien, 2022; ISBN print 978-3-903311-19-0, TU Verlag

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

Energy-related Material Flow Simulation in Production and Logistics
S. Wenzel, M. Rabe, S. Straßburger, C. von Viebahn (Hrsg.), Springer Nature 2023; ASIM Mitteilung AM 182

Simulation in Production and Logistics 2021 – 19. ASIM Fachtagung Simulation in Produktion und Logistik
Online Tagung, Sept. 2021, J. Franke, P. Schuderer (Hrsg.), Cuvillier Verlag, Göttingen, 2021,
ISBN print 978-3-7369-7479-1; ISBN ebook 978-3-7369-6479-2; ASIM Mitteilung AM177

Proceedings ASIM SST 2020 – 25. ASIM Symposium Simulationstechnik, Online-Tagung
14.-15.10.2020; C. Deatcu, D. Lückerath, O. Ullrich, U. Durak (Hrsg.), ARGESIM Verlag Wien, 2020;
ISBN ebook: 978-3-901608-93-3; DOI 10.11128/arep.59; ARGESIM Report 59; ASIM Mitteilung AM 174

Simulation in Production and Logistics 2019 – 18. ASIM Fachtagung Simulation in Produktion und Logistik
Chemnitz, 18.-20. 9. 2019; M. Putz, A. Schlegel (Hrsg.), Verlag Wissenschaftliche Skripten Auerbach, 2019,
ISBN print 978-3-95735-113-5, ISBN ebook 978-3-95735-114-2; ASIM Mitteilung AM172

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

Tagungsband ASIM SST 2018 - 24. ASIM Symposium Simulationstechnik, HCU Hamburg, Oktober 2018
C. Deatcu, T. Schramm, K. Zobel (Hrsg.), ARGESIM Verlag Wien, 2018; ISBN print: 978-3-901608-12-4;
ISBN ebook: 978-3-901608-17-9; DOI 10.11128/arep.56; ARGESIM Report 56; ASIM Mitteilung AM 168

Book Series Fortschrittsberichte Simulation – Advances in Simulation

Aufgabenorientierte Multi-Robotersteuerungen auf Basis des SBC-Frameworks und DEVS. Birger Feymann, FBS 40,
ISBN ebook 978-3-903347-40-3, DOI 10.11128/fbs.40, ARGESIM Publ. Vienna,2022; ISBN print 978-3-903311-20-6, TUVerlag Wien, 2022

Cooperative and Multirate Simulation: Analysis, Classification and New Hierarchical Approaches. Irene Hafner, FBS 39
ISBN ebook978-3-903347-39-7, DOI 10.11128/fbs.39, ARGESIM Publ. Vienna,2022; ISBN print978-3-903311-07-7, TUVerlag Wien, 2022

Die Bedeutung der Risikoanalyse für den Rechtsschutz bei automatisierten Verwaltungsstrafverfahren. T. Preiß, FBS 38
ISBN ebook 978-3-903347-38-0, DOI 10.11128/fbs.38, ARGESIM Publ. Vienna,2020; ISBN print 978-3-903311-14-5, TUVerlag Wien, 2020

Methods for Hybrid Modeling and Simulation-Based Optimization in Energy-Aware Production Planning. B. Heinzl, FBS 37
ISBN ebook 978-3-903347-37-3, DOI 10.11128/fbs.37, ARGESIM Publ. Vienna,2020; ISBN print 978-3-903311-11-4, TUVerlag Wien, 2020

Konforme Abbildungen zur Simulation von Modellen mit verteilten Parametern. Martin Holzinger, FBS 36
ISBN ebook 978-3-903347-36-6, DOI 10.11128/fbs.36, ARGESIM Publ. Vienna, 2020; ISBN print 978-3-903311-10-7, TUVerlag Wien, 2020

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

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

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

Deployment of Mathematical Simulation Models for Space Management. Stefan Emrich, FBS 22;
ISBN ebook 978-3-903347-22-9, DOI 10.11128/fbs.22, ARGESIM Publisher Vienna, 2016

Lattice Boltzmann Modeling and Simulation of Incompressible Flows in Distensible Tubes for Applications in Hemodynamics.
X. Descovich, FBS 21; ISBN ebook 978-3-903347-21-2, DOI 10.11128/fbs.21, ARGESIM, 2016; ISBN Print 978-3-903024-98-4, TUVerlag 2019

Mathematical Modeling for New Insights into Epidemics by Herd Immunity and Serotype Shift. Florian Miksch, FBS 20;
ISBN ebook 978-3-903347-20-5, DOI 10.11128/fbs.20, ARGESIM Publ. Vienna, 2016; ISBN Print 978-3-903024-21-2, TUVerlag Wien, 2016

Integration of Agent Based Modelling in DEVS for Utilisation Analysis: The MoreSpace Project at TU Vienna. S. Tauböck, FBS 19
ISBN ebook 978-3-903347-19-9, DOI 10.11128/fbs.19, ARGESIM Publ., 2016; ISBN Print 978-3-903024-85-4, TUVerlag Wien, 2019

Ein Beitrag zur prädiktiven Regelung verbrennungsmotorischer Prozesse. Christian Steinbrecher, FBS 18
ISBN ebook 978-3-903347-18-2, DOI 10.11128/fbs.18, 2018; ISBN Print 978-3-901608-68-1, 2011; ARGESIM Publisher

Simulation Based Parameter and Structure Optimisation of Discrete Event Systems. Olaf Hagendorf, FBS 17
ISBN ebook 978-3-903347-17-5, DOI 10.11128/fbs.17, 2019; ISBN Print 978-3-901608-67-4, 2010; ARGESIM Publisher

Simulation of Arterial Blood Flow with the Lattice Boltzmann Method. Daniel Leitner, FBS 16
ISBN ebook 978-3-903347-16-8, DOI 10.11128/fbs.16, 2020; ISBN Print 978-3-901608-66-7, 2009; ARGESIM Publisher

Download via ASIM - **www.asim-gi.org** **Print-on-Demand** via TUVerlag **www.tuverlag.at**

EUROSIM 2023:

Simulation for a Sustainable Future

July 3-5, 2023, Amsterdam, The Netherlands

The 11th edition of the EUROSIM Congress is dedicated to a sustainable future. And there is no better place than Amsterdam for celebrating the link between simulation and sustainability. You will be pleased to discover the fascinating and vibrant city of Amsterdam, a hub for many sectors such as logistics, ICT, art, financial and business services, fashion, and many others.

Authors are welcome to submit their contributions to the following, but not limited to areas such as:
Agent-Based Simulation ● Methodology ● Supply Chain Management, Logistics & Transportation ● Industrial Case Studies ● Complex, Intelligent, Adaptive and Autonomous Systems (CIAAS) ● Data Science for Simulation ● Environment and Sustainability Applications ● Gaming ● Healthcare Applications ● Hybrid Simulation ● Manufacturing Applications ● Military Applications ● Simulation in Education ● Simulation-Optimization ● Advanced Tutorials/workshops from academics and industry, PhD Track and many more

Paper deadlines and requirements:

The congress encourages researchers and practitioners to submit their original contributions as:

- Discussing paper (2-page extended abstract); peer-reviewed; accepted for presentation
- Full paper, (max 8 pages); peer-reviewed; accepted for presentation; published in the Congress proceedings; considered for journal publication.

Instructions, information, submission forms, and procedures are available on the EUROSIM 2023 website ***www.eurosim2023.eu***. Each accepted contribution requires a paid registration.

Important dates:

● Extended abstract/ Full paper submission deadline:	Begin 2023
● Notification of acceptance:	28 February 2023
● Final paper submission deadline:	31 March 2023
● Early bird registration deadline:	30 April 2023
● Last call for registration deadline:	31 May 2023
● EUROSIM Congress 2023:	28-30 June 2023

Submit your paper at: ***https://easychair.org/conferences/?conf=eurosim2023***

www.tuverlag.at
ISBN 978-3-903311-33-6