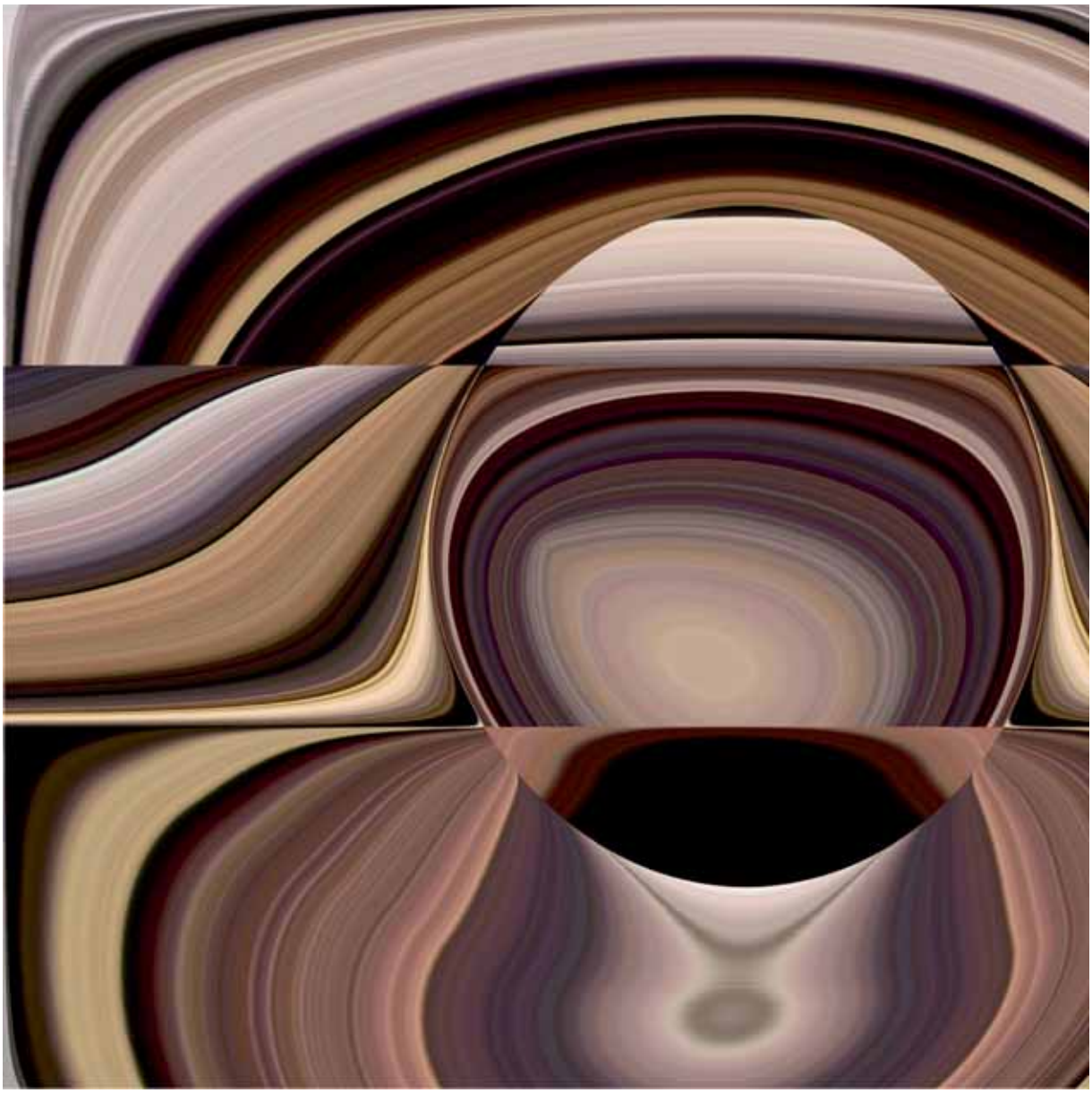
SNE SIMULATION NOTES EUROPE



Volume 25 No.1 April 2015 doi: 10.11128/sne.25.1.1027

Journal on Developments and Trends in Modelling and Simulation

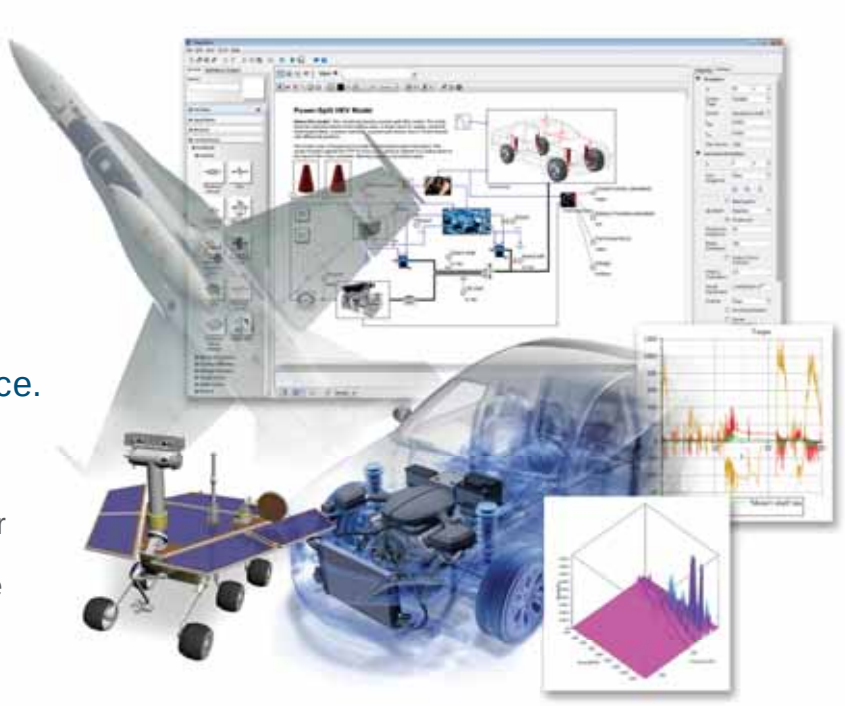
Membership Journal for Simulation Societies and Groups in EUROSIM

Print ISSN 2305-9974
Online ISSN 2306-0271

ARGESIM

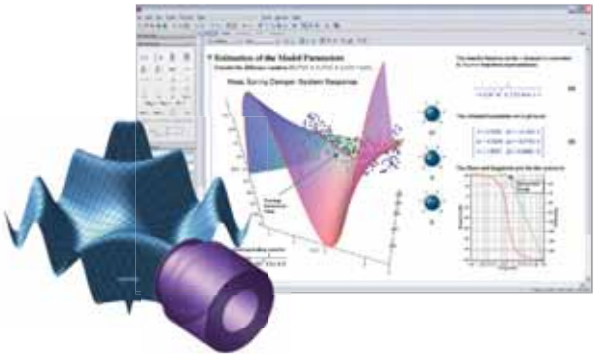
A modern approach to modeling and simulation

MapleSim™



With MapleSim, educators have an industry-proven tool to help bridge the gap between theory and practice.

- MapleSim illustrates concepts, and helps students learn the connection between theory and physical behavior
- A wide variety of models are available to help get started right away



Maple™

MapleSim is built on Maple, which combines the world's most powerful mathematical computation engine with an intuitive, "clickable" user interface.

To learn more about how you can reinforce engineering concepts using a combination of theory, simulation, and hardware, view this webinar.

www.maplesoft.com/SNEWebinar

Contact us: +49 (0)241/980919-30

Maplesoft
Mathematics • Modeling • Simulation
A Cybernet Group Company

www.maplesoft.com | germany@maplesoft.com

Editorial

Dear Readers – *We are glad, that for SNE Volume 25 again Vlatko Ceric, past president of CROSSIM, is providing his algorithmic art as design for SNE cover pages. 'Algorithms, mathematics and art are interrelated in an art form called algorithmic art. Algorithmic art is visual art generated by algorithms that completely describe creation of images. This kind of art is strongly related with contemporary computer technology, and especially computer programming, as well as with mathematics used in algorithms for image generation' – as Vlatko Ceric defines. The technique used for the picture series for the covers of SNE Volume 25 is alienation of 'classic' pictures by certain algorithms (details in next issues).*
The content of the scientific part reflects the broad variety of nowadays modeling and simulation. 'Discrete' techniques for modelling and simulation generate robust schedules for tramway networks, increase space utilization in office buildings, support server outage detection, and compare pedestrian flow models on a microscopic level. A case study compares modeling of convection and diffusion, and an Education Note discusses simulation system for radiology education integration of physical and virtual realities. And last but not least, this issue continues the ARGESIM Benchmark Series, with a solution to Benchmark C8 'Canal and Lock System' and with a revised definition of Benchmark C17 'Modelling and Simulation of a SIR-type Epidemic with Cellular Automata and Differential Equations', which allows a much butter comparison both approaches.

I would like to thank all authors for their contributions, and the organizers of the EUROSIM conferences for co-operation in post-conference publication, and the ARGESIM SNE staff for helping to manage the SNE administration and the improved SNE layout and extended templates for submissions (now also tex).

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

Contents SNE 25(1)

SNE doi: 10.11128/sne.25.1.1027

Reader's Info

Simulation Notes Europe publishes peer reviewed *Technical Notes*, *Short Notes* and *Overview Notes* on developments and trends in modelling and simulation in various areas and in application and theory, with main topics being simulation aspects and interdisciplinarity.

Individual submissions of scientific papers are welcome, as well as post-conference publications of contributions from conferences of **EUROSIM** societies.

Furthermore **SNE** documents the **ARGESIM Benchmarks** on *Modelling Approaches and Simulation Implementations* with publication of definitions, solutions and discussions (*Benchmark Notes*). Special *Educational Notes* present the use of modelling and simulation in and for education and for e-learning.

SNE is the official membership journal of **EUROSIM**, the Federation of European Simulation Societies. A News Section in **SNE** provides information for **EUROSIM** Simulation Societies and Simulation Groups.

SNE is published in a printed version (Print ISSN 2305-9974) and in an online version (Online ISSN 2306-0271). With **Online SNE** the publisher **ARGESIM** follows the **Open Access** strategy, allowing download of published contributions for free. Since 2012 **Online SNE** contributions are identified by a DOI (Digital Object Identifier) assigned to the publisher **ARGESIM** (DOI prefix 10.11128).

Print SNE, high-resolution **Online SNE**, full **SNE Archive**, and source codes of the *Benchmark Notes* are available for members of **EUROSIM** societies.

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

SNE Issue 25(1) April 2015 doi: 10.11128/sne.25.1.1027

→ *www.sne-journal.org*

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

SNE Editorial Office, c/o ARGESIM / DWH,
Neustiftgasse 57-59, 1070 Vienna, Austria

SNE Editorial Board

Author's Info

Authors are inv ited to subm it contributions which have not been published and have not being considered for publication elsewhere to the **SNE** Ed itorial Office. Further more, SNE invites org anizers of EUROSIM confer ences to submit post-conference publication for the authors of their conferences.
SNE distinguishes different types of contributions (*Notes*):

- *Overview Note* – State-of-the-Art report in a specific area, up to 14 pages, only upon invitation
- *Technical Note* – scientific publication on specific topic in modelling and simulation, 6 – 8 pages
- *Education Note* – modelling and simulation in / for education and e-learning; 6 - 8 pages
- *Short Note* – recent development on specific topic, max. 4 p.
- *Software Note* – specific implementation with scientific analysis, max 4 pages
- *Benchmark Note* – Solution to an ARGESIM Benchmark; commented solution 4 pages, comparative solutions 4-8 pages

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

SNE Contact & Info

→ *www.sne-journal.org*
✉ *office@sne-journal.org, eic@sne-journal.og*
✉ SNE Editorial Office, ARGESIM / dwh Simulation Services, Neustiftgasse 57-59, 1070 Vienna, Austria

SNE SIMULATION NOTES EUROPE

ISSN SNE Print ISSN 2305-9974, SNE Online ISSN 2306-0271
WEB: → *www.sne-journal.org*, DOI prefix 10.11128/sne
Scope: Technical Notes, Short Notes and Overview Notes on developments and trends in modelling and simulation in various areas and in application and theory; benchmarks and benchmark documentations 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, Vienna Univ. of Technology, Inst. f. Analysis and Scientific Computing, Div., Math. Modelling and Simulation, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria; ✉ *Felix.Breitenecker@tuwien.ac.at,* ✉ *eic@sne-journal.org*
Layout / Administration: J. Tanzler, F. Preyser, T. Vobruba; C. Wytrzens, R. Leskovar et al.; Math. Modelling and Simulation Group, Vienna Univ. of Technology, Wiedner Haupstrstasse 8-10, 1040 Vienna , ✉ *office@sne-journal.org*
Print SNE: Grafisches Zentrum, TU Vienna, Wiedner Hauptstrasse 8-10, 1040, Vienna, Austria
Online SNE: ARGESIM /ASIM, c.o. dwh Simulation Services, Neustiftgasse 57-59, 1070 Vienna, Austria
Publisher: ARGESIM ARBEITSGEMEINSCHAFT SIMULATION NEWS - WORKING COMMITTEE SIMULATION NEWS, Neustiftgasse 57-59, 1070 Vienna, Austria; → *www.argesim.org,* ✉ *info@argesim.org* on behalf of ASIM(→ *www.asim-gi.org* and EUROSIM → *www.eurosim.info*

A Robust Schedule for Montpellier's Tramway Network

Oliver Ullrich[1*], Daniel Lückerath[2], Ewald Speckenmeyer[2]

[1]National Science Foundation's Industry-University Cooperative Research Center, Florida International University, ECS 243C, 11200 SW 8th St, Miami FL-33199; *oullrich@fiu.edu

[2]Institut für Informatik, Universität zu Köln, Albertus-Magnus-Platz, 50923 Köln, Germany

Simulation Notes Europe SNE 25(1), 2015, 1 - 8
DOI: 10.11128/sne.25.tn.10271
Received: September 28, 2014 (Selected ASIM SST 2014 Postconf. Publ.); Accepted: January 10, 2015;

Abstract. The city of Montpellier in the Languedoc-Roussillon region of France features a fast growing tram network as a central part of its public service infrastructure. Here, as in many other tram networks, resources like tracks and stations are shared between different lines. Because of the resulting dependencies, small inevitable delays can spread through the network and affect its global performance.

This article examines whether a robust tram schedule may help to raise punctuality in Montpellier's tram network. To accomplish this, we apply a tool set designed to generate schedules optimized for robustness, which also satisfy given sets of planning requirements. These tools allow to compare time tables with respect to their punctuality and other key indicators.

After an introduction to the goals of this paper, we continue with a description of the tool set focusing on optimization and simulation modules. These software utilities are then employed to generate and simulate robust and non-robust schedules for Montpellier's tram network, which are subsequently compared for the resulting delays.

Introduction

The city of Montpellier in southern France is growing fast, its population has tripled in the last fifty years ([6]). As major part of the city's public service infrastructure, the Tramway tram network is provided by Transports de l'agglomération de Montpellier (TAM). The first Tramway line was launched in 2000, it connects the eastern and western suburbs to the city center.

Since then three more lines commenced operation. By now, about 282,000 passengers are served on each weekday (see [17]), which amounts to about half of the population of Montpellier's metropolitan area. Three more tram lines are commissioned, the first of which is planned to commence operation in 2017.

In Montpellier's tram network, several lines share resources like platforms, switches and tracks. Because of the resulting dependencies, small local delays can propagate to succeeding trams, build up to larger delays, and thus affect the network's global performance.

In this paper, we explore whether a robust schedule can help to reduce delays in Montpellier's Tramway network. We define robustness as the degree to which inevitable small delays are kept local to the immediately affected tram and do not spread through the network. To examine this, we apply a software tool chain which enables us to generate robust schedules, compare their feasibility and evaluate their punctuality and other key indicators.

This paper continues with a description of our approaches on optimization and simulation of tram schedules (Section 1). It then focuses on the modelling and simulation of Montpellier's Tramway system. Robust and non-robust schedules are generated, simulated, and compared concerning the resulting delays (Section 2). The paper closes with a short summary of lessons learned and some thoughts on further research (Section 3).

1 Simulation and Optimization of Tram Schedules

The project "Computer Aided Traffic Scheduling" (CATS) is built around a database complying with the ÖPNV5 data model proposed by the association of German transport companies (Verband Deutscher Verkehrsunternehmen, see [23]).

Visualization, optimization, and simulation modules are connected via operations on the database and through XML configuration files (see Figure 1). Due to its compliance with the ÖPNV5 data model our framework is capable of working on many European tram networks.

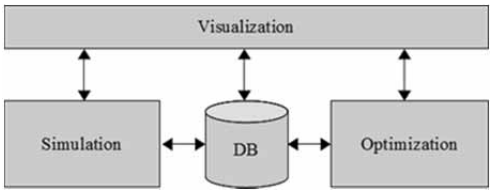


Figure 1: Architecture of project CATS.

For an in-depth description of the optimization method, see [21]. A more detailed discussion of the simulation software can be found in [7]. (This section is an abbreviated version of [21], Section 1.)

1.1 Optimization of tram schedules

Various approaches to optimize tram and railway schedules are known (see e.g. [1, 3, 4, 5, 9, 10, 12]). Most of them aim at one general objective like minimizing vehicle delay (see [10, 12]) or maximizing robustness to restrict the global impact of small, local disturbances (see [4, 5]). Others use a combination of objectives, like operational profit and robustness in [3], or combining social opportunity cost and operational cost in [11].

Because of the complex nature of the problem, many authors use heuristic approaches like Lagrangian heuristics (see [3]) or simulated annealing (see [11]). Others (see [1]) introduce exact algorithms for restricted subclasses, like chain and spider networks.

In our project, we combine heuristics and exact methods to generate optimal synchronized time tables for tram networks, targeting maximal robustness and adherence to transport planning requirements at the same time. Those planning requirements originate from political, economic and feasibility reasons. Thus it is not sufficient to exclusively consider a general goal like robustness when generating time tables.

We use the scheduled time offset between two consecutively departing vehicles at a platform as an indicator for robustness. In an assumed tact interval of ten minutes, two lines could be scheduled with equidistant offsets of five minutes, which means that one or both involved vehicles could be late for more than four minutes without consequences for the following tram.

Under an extremely unequal split of the available time span into a nine minute offset followed by a one minute offset, the first tram could have a delay of more than eight minutes without consequences to the following vehicle. On the other hand, would the second vehicle be even slightly late, the delay would spread to the follow-up tram. Since we are assuming typically small delays, we see an equidistant distribution as very robust, the occurrence of very small offsets as not robust.

So, to calculate the robustness of a time table λ we examine at each platform h of the network the scheduled time offset $\delta_{f,pred(f)}(h,\lambda)$ between any trip f and its predecessor $pred(f)$, i.e. the time elapsed between the departures of $pred(f)$ and f at platform h.

To reduce complexity we aggregate subsequent similar platforms operated by the same lines to a maximal platform type h', weighted by the number of included platforms $\varphi_{h'}$ (see Figure 2). The reduced set of platforms is denoted by H'.

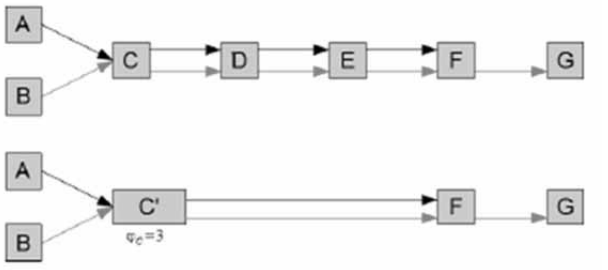


Figure 2: Example for platform reduction.

To calculate the robustness Φ_a of schedule λ, we add the inverse of $\delta_{f,pred(f)}(h',\lambda)$ for each platform type $h' \in H'$ and all its trips, thus applying a penalty for small safety distances. With $f \in F_{h'}$ representing all trips that serve platform type h' under schedule λ, and $\varphi_{h'}$ as the number of platforms represented by h', the resulting function is as follows:

$$\Phi_a(\lambda) = \sum_{h' \in H'} \sum_{f \in F_{h'}} \frac{1}{\delta_{f,pred(f)}(h',\lambda)} * \varphi_{h'} \qquad (1)$$

Given is a set R of planning requirements, with r denoting a single requirement $r \in R$. In order to calculate the compliance with transport planning requirements we introduce $\rho_r(\lambda) \in \{1,2,3,\infty\}$ the compliance factor of requirement r under a schedule λ. A compliance factor of 1 means that the requirement is completely satisfied, 2 and 3 denote tolerable compliance, and ∞ means that the constraint is not met and the time table candidate λ must be rejected.

We add the compliance values for all $r \in R$ and get the following:

$$\Phi_b(\lambda) = \sum_{r \in R} \rho_r(\lambda) \quad (2)$$

Depending on the network under consideration and the number of planning requirements, the two parts of the objective function may not be comparable directly. Thus we define a normalizing factor σ, which reflects the relationship between the lower bounds of both parts. The theoretically optimal distance $\delta^{opt}_{f,pred(f)}(h')$ of two trips $pred(f)$ and f on platform type h' is obtained by dividing the tact interval by the number of serving lines at that platform type. The best possible compliance factor ρ_r^{min} of a planning requirement $r \in R$ is the minimal value assigned by the planner, independent of the characteristics of the examined solution candidate. Typically $\rho_r^{min} = 1$. We define σ as:

$$\sigma = \sum_{h' \in H'} \sum_{f \in F_{h'}} \frac{1}{\delta^{opt}_{f,pred(f)}(h')} \Big/ \sum_{r \in R} \rho_r^{min} \quad (3)$$

Combining $\Phi_a(\lambda)$ and $\Phi_b(\lambda)$ yields the overall objective function $\Phi(\lambda)$ (see Equation 4), normalized by σ and weighted by a factor $0 \leq \alpha \leq 1$, the relative weight of the fulfillment of planning requirements.

$$\Phi(\lambda) = (1-\alpha) * \sum_{h' \in H'} \sum_{f \in F_{h'}} \frac{1}{\delta_{f,pred(f)}(h', \lambda)} * \varphi_{h'} + \alpha * \sigma * \sum_{r \in R} \rho_r(\lambda) \quad (4)$$

In our experiments, this weight is set to $\alpha = 0.5$, so that robustness and the fulfillment of requirements are equally important.

A valid solution also has to adhere to some other constraints. The first restriction requires each start time μ_i of each line variant i to be inside the tact interval, with $t_{interval}$ being the duration of the interval (see Equation 5).

$$\forall i \leq |\lambda| : 0 \leq \mu_i < t_{interval} \quad (5)$$

Another restriction requires an offset of at least one minute between two departures of the trips f and $pred(f)$ at each platform type $h' \in H'$(see Equation 6). This means that no platform can be blocked by more than one train at any point of time, the schedule has to be free of collisions.

$$\forall h' \in H' : \forall f \in F : \delta_{f,pred(f)}(h', \lambda) > 0 \quad (6)$$

We identify seven types of transport planning constraints: Interval constraints, start time constraints, core line constraints, bidirectional track constraints, turning point constraints, warranted connection constraints and follow-up connection constraints. Upon closer inspection (see [20], Section 6.2.3) it becomes clear that interval and start time constraints are fundamental and all other constraint types can be expressed using these two. E.g. a bidirectional track constraint can be expressed by two interval constraints covering opposing platforms. Subsequently only interval and start time constraints are considered in the remainder of this paper.

The presented model is implemented as a branch-and-bound solver, which starts with an initial solution computed by a genetic algorithm for performance reasons. For implementation details see [21] or [20], Sections 6.2.4 and 6.2.5.

1.2 Simulation of tram schedules

Most rail-bound traffic simulation models are designed for long distance train or railway networks, see e.g. [8, 9]. While those systems feature similarities to tram networks, e.g. passenger exchange or maneuvering capabilities, they differ significantly in important aspects. Tram networks are often mixed, i.e. trams travel on underground tracks as well as on street level, and are thus subject to individual traffic and corresponding traffic regulation strategies. Subsequently, tram behavior is a mixture between train and car behavior, e.g. line-of-sight operating/driving. Therefore a simple adaption of railway simulation methodologies is not feasible.

Our application is based upon a model-based parallelization framework (described in [20] and [22]), which exploits the embedded model's intrinsic parallelism. The mixed tram network is modeled as a directed graph with platforms, tracks and track switches represented by nodes. Connections between nodes are represented as edges. The distributions for the duration of passenger exchange are specific to platform and tram type with the combined duration of opening and closing the vehicle doors as minimum value. Vehicles encapsulate most of the simulation dynamics, which are based upon the event based simulation approach (as described in [2]). Thus trams change their state at events of certain types, like stopping or accelerating, which happen at discrete points in time. These state changes may trigger a change in the over-all system state and generate follow-up events, which are administrated in a priority queue.

Tram attributes are specified by the type of tram, which holds functions for the maneuvering capabilities, e.g. acceleration and braking. Main parameters of the simulation are the maximum driving velocity v_{max}, the dawdle probability $0 \leq p_d \leq 1$ (which maps the chance that a tram's driver does not accelerate at a given moment due to external causes), and the dawdle factor $d > 1$ (which maps the amount of the delay caused by dawdling). For the experiments, these values are set as $p_d = 0.3$ and $d = 1.3$. A more detailed description of model and implementation can be found in [7].

2 Examining Montpellier's Tramway Network

We apply the described software suite to Montpellier's Tramway network (for an overview see Figure 3) based on the time table data of 2013 (gathered from [13], [14], [15], and [16]). The system consists of 84 stations with 176 platforms and 46 track switches, connected via 232 tracks (see [19]). These tracks cover a total length of ca. 56 kilometers, resulting in an average track length of about 241 meters. 1,215 trips per operational day are executed on four lines with 24 line routes (see Figure 4), about 282,000 passengers are served on each weekday (see [17]).

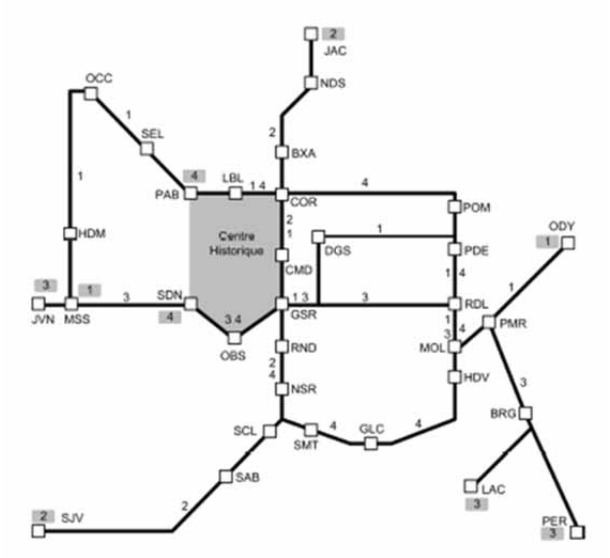


Figure 3: Montpellier's Tramway network.

Figure 4: Montpellier's line routes.

2.1 Schedule generation

The schedule implemented by TAM has no global tact interval, trains serve the routes in varying patterns through the day. At peak times, lines 1 and 2 are traversing the city center every four to five minutes, with changing headway. Line 3 is served every six to eight minutes, the intervals between consecutive trains of line 4 are alternating between eight and nine minutes. To find an appropriate approximation of this, we assume a tact interval of eight minutes, and insert additional core lines 1A and 2A to double the frequency of lines 1 and 2 to four minutes. A set of planning requirements is defined, which can be decomposed to 16 interval constraints. These include the additional core lines 1A and 2A, and minimum turn-around times at line ends.

The genetic algorithm is initialized with a population of 450 randomly generated individuals. The best fitness value of this first generation is 75.55 (average: 83.58, worst value: 95.00). In the course of 500 generations and a runtime of 313 seconds, the algorithm finds a best solution candidate with a fitness value of 75.25 (average: 75.51, worst value: 80.11). The branch-and-bound solver further enhances the minimal fitness value in the course of a 200 seconds run down to 75.22, and finds 128 optimum solutions.

2.2 Comparing generated schedules

We pick ten schedules each out of both the pool of initial solutions and the optimum solution pool and execute ten simulation runs for each of those 20 schedules. The maximum velocity is set to $v_{\max} = 40$ km/h, a compromise between the observed inner city maximum speed of 30 km/h and the higher speed in some regions outside the city with exclusive track usage.

The runs under the initial schedules yield an average delay of all departures of 9.8 seconds. Under the best schedules the average delay is 8.2 seconds, which means a reduction of 16.3 percent or 1.6 seconds. The average delay of all delayed departures is reduced from 25.8 by 2.3 seconds or 8.9 percent to 23.5 seconds.

The frequency distribution of occurring delays was also collected (see Figure 5). Under the optimal schedules, the numbers of delays in each bucket are reduced. This effect is especially significant for the larger delays of more than 60 seconds (see Figure 6). The total number of departures with a larger delay is reduced from 521.3 under the random schedules by 210.7 departures or 40.4 percent down to 310.6.

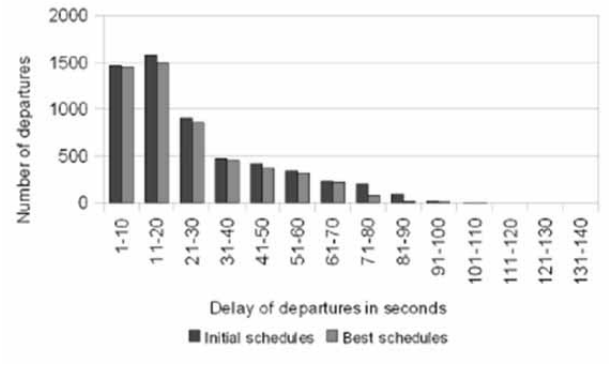


Figure 5: Frequency distribution of delays.

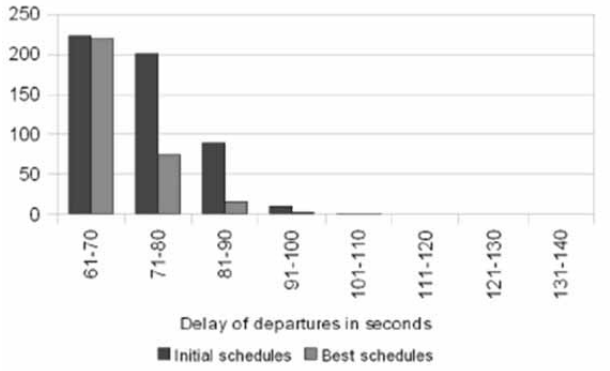


Figure 6: Frequency distribution of delays.

As seen, robust schedules reduce the average delay in the Tramway network, though only by a small amount, and significantly reduce the number of larger delays. Under optimum schedules with their better distributed time offsets, many small delays can be made up for fast and do not spread to consecutive departures. A higher robustness can thus help to reduce the number of larger delays by preventing inevitable small delays from accumulating over the simulation run.

To take a more detailed look at the model's behavior, we pick a typical schedule A (see Table 1) with an objective function value of 92.69 from the genetic algorithm's initial pool of valid solution candidates, and a schedule B (see Table 2) from the pool of best solutions. We examine both schedules by executing 100 simulation runs each and comparing the results.

Line/Route	1	1A	2	2A	3	4
01	0	6	3	1	7	3
02	5	3	3	1	6	2

Table 1: Scheduled departures at the routes' starting points under schedule A.

Line/Route	1	1A	2	2A	3	4
01	6	1	4	0	1	5
02	3	7	7	3	0	5

Table 2: Scheduled departures at the routes' starting points under schedule B.

On average, schedule A yields a line delay of 8.7 seconds, which is reduced under schedule B by 17.2 percent or 1.5 seconds to 7.3 seconds. The only line that yields a significantly lower delay under the optimum schedule is line 2, with a reduction of 24.3 percent from 21.6 to 16.3 seconds (see Figure 7).

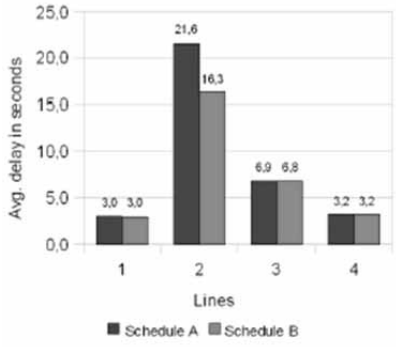


Figure 7: Average delay of lines.

To examine this, we take a closer look at trips 3 and 4 of tram 2005 (see Figure 8), which serves the shorter routes 205 and 206 of line 2A. While the measured delays at several platforms vary, the most obvious differences are found in the regions of the town center around Corum (COR, see Figure 3) and Gare Saint-Roch (GSR).

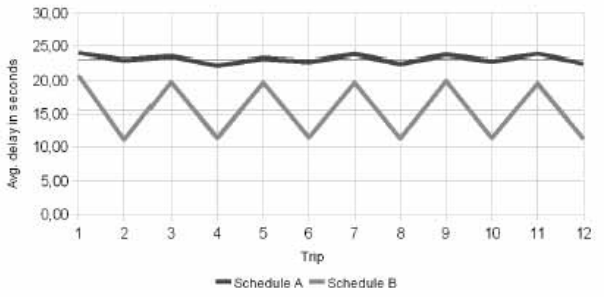


Figure 8: Average trip delays of tram 2005 serving line 2A.

Serving trip 3 in the direction of Sabines (see Figure 9), trams of line 2A enter an array of switches they share with lines 1, 1A, 2 and 4 after the departure at Corum. Under schedule A, the vehicle has to wait to access these common resources, and cannot regain the resulting delay until after the stop at Nouveau Saint-Roch (NSR). Under schedule B with its better distributed time offset, these resources are instantly accessible to the tram.

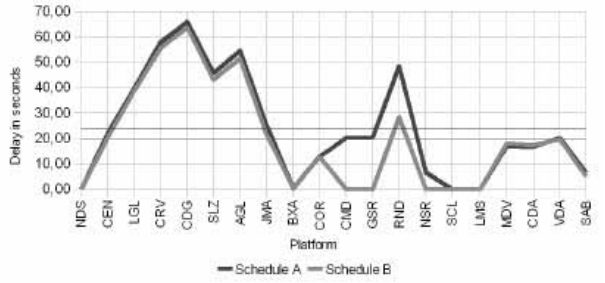


Figure 9: Average delays at platforms of trip 3 of tram 2005.

On the return trip in the direction of Notre-Dame de Sablassou (see Figure 10), the tram has to navigate four consecutive switches between the stations Rondelet (RND) and Gare Saint-Roch. It shares some of these switches with all other lines. Under the random schedule A, the vehicle gets behind a tram serving line 1, although it is scheduled to precede it by one minute. It therefore has to wait for that tram to clear the Gare Saint-Roch platform and thus gets a delay of about 80 seconds. It can start to regain the delay after lines 1 and 2 split course before Corum station.

As described, only line 2 (and its companion line 2A) shows a significantly lower delay under the robust schedule B, the other lines yield the same values under both schedules. Lines 1 and 4 run in parallel for a while, then part way and rejoin after sections of different track lengths and planned driving times (see Figure 3).

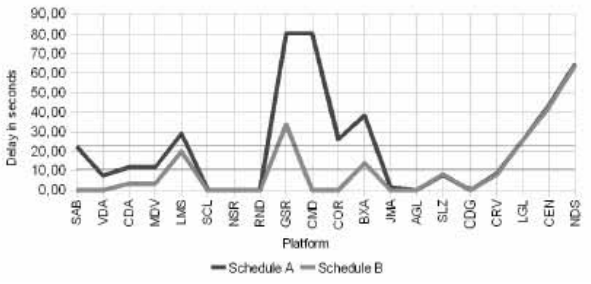


Figure 10: Average delays at platforms of trip 4 of tram 2005.

Because of this, and considering that the optimizer can only generate valid schedules with a time offset of at least one minute at each platform, these lines are locked in relation to each other under all valid schedules. There is no way the optimizer can generate a better (or worse) schedule concerning these two lines. The same applies to the combination of lines 3 and 4: These are also locked under any valid schedule.

This phenomenon does not occur with line 2, which runs parallel to lines 1 and 4, but only in one contiguous section each. It does then split from these lines but does not rejoin them later. These lines are therefore not locked, the optimizer can schedule line 2 more freely.

The experiments show that the application of a robust schedule can help to reduce delays in Montpellier's tram network. They also show that robustness has its main impact in those regions of the networks where resources are shared by most line routes. In the Tramway network these regions are switch arrays near the stations Gare Saint-Roch and Corum.

2.3 TAM's applied schedule

To complete the picture of Montpellier's Tramway network, we also examine the schedule applied by TAM at the time of this writing. Because it adheres to no common tact interval and comprehends planning requirements unknown to the authors, the results cannot be compared directly to the generated schedules. Therefore, no insights about special traffic phenomenons should be assumed.

As described in Section 2.1, TAM's schedule has no common tact interval. Therefore, the numbers of the started trips per hour deviate in the sample period of 08.00 to 16.59 (see Figure 11) from their counterparts of the generated schedules.

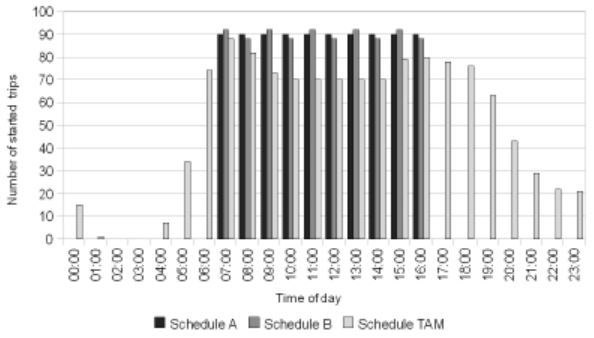


Figure 11: Number of started trips per hour.

The data gathered by executing 100 simulation runs with the described parameters shows TAM's schedule to be in general range with the generated schedules. The average delay of departures of 8.1 seconds is slightly smaller than the value yielded by schedule B, and 1.7 seconds smaller than that of schedule A. The average delay of delayed departures has a value of 24.4 seconds and is therefore splitting the distance between schedule A (25.8 seconds) and schedule B (23.5 seconds). The number of larger delays is 314.4, on about the level of schedule A's value.

The frequency distribution shows that TAM's schedule yields a lower number of small delays, which are compensated by a higher number of delays of more than 70 seconds (see Figure 12).

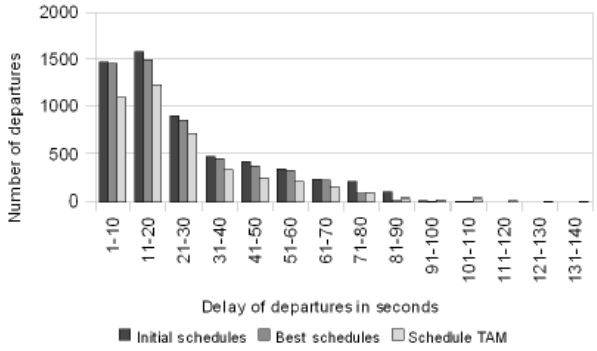


Figure 12: TAM's applied schedule - Frequency distribution.

TAM's schedule yields line delays which are comparable to the values resulting from the generated schedules (see Figure 13): line 1 has the same value under all three schedules, line 2 lies between the values of schedules A and B, lines' 3 and 4 delay values are a bit higher than their counterparts.

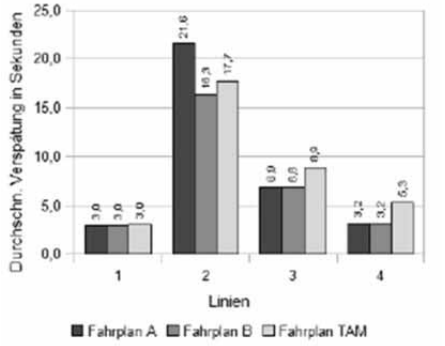


Figure 13: TAM's applied schedule - Average delay of lines.

3 Conclusions and Further Research

This article showed an approach to examine the influence of robustness on a tram network. To accomplish this, we applied optimization and simulation tools designed to evaluate schedules optimized for robustness. These software utilities were employed to generate and compare robust and non-robust schedules for Montpellier's tram network, demonstrating that a robust schedule can indeed help to reduce delays in the Tramway network. The experiments showed that the main improvements center in those regions of the networks where resources are shared by most line routes. In the presented case these regions are the switch arrays near the stations Gare Saint-Roch and Corum.

Montpellier's Tramway network is expanding: a line 5 is currently being built and will connect the fast growing suburbs in the north and west to the inner city (see [18]). Supporting a rerouted line 4, this line will complete the ring track around the historical city center. Line 5 is planned to commence operation in 2017. The city of Montpellier already commissioned lines 6 and 7, their exact routes are still under consideration. Our model will be expanded with representations of these lines, the resulting model's behavior will be analysed and compared to the existing models.

We also plan to analyze under which general circumstances a robust schedule will increase punctuality in a tram network. The presented results of Montpellier's Tramway and of our hometown Cologne's KVB network (see [21]) will be utilized as a base for this.

Acknowledgements. This material is partially based upon work supported by the National Science Foundation under grants I/UCRC IIP-1338922, AIR IIP-1237818, SBIR IIP-1330943, III-Large IIS-1213026, MRI CNS-0821345, MRI CNS-1126619, CREST HRD-0833093, I/UCRC IIP-0829576, MRI CNS-0959985, and FRP IIP-1230661 and U.S. Department of Transportation under a TIGER grant.

References

[1] Bampas E, Kaouri G, Lampis M, Pagourtzis A. Periodic Metro Scheduling. In: Jacob R, Müller-Hannemann M, editors. ATMOS, 2006. Proceedings of the ATMOS 2006 - 6th Workshop on Algorithmic Methods and Models for Optimization of Railways; Internationales Begegnungs und Forschungszentrum für Informatik (IBFI;, 2006.

[2] Bank J, Carson JS, Nelson BL, Nicol DM. *Discrete-Event System Simulation*. Pearson, Prentice Hall; 2010. 640 p.

[3] Cacchiana V, Caprara A, Fischetti M. A Lagrangian Heuristic for Robustness, with an Application to Train Timetabling. *Transportation Science*. 2012; 46(1): p 124-133.

[4] Caimi G, Fuchsberger M, Laumanns M, Schüpbach K. 2010. Periodic Railway Timetabling with Event Flexibility. *Networks*. 57(1): p 3-18.

[5] Genç Z. 2003. Ein neuer Ansatz zur Fahrplanoptimierung im ÖPNV: Maximierung von zeitlichen icherheitabständen [dissertation]. [Mathematisch-Naturwissenschaftliche Fakultät]: Universität zu Köln.

[6] Institut national de la statistique et des études économiques: La population de Montpellier Agglomération a triplé au cours des cinquante dernières années [Internet]. [cited 2013 June 26]. Available from: http://www.insee.fr/fr/themes/document.asp?ref_id=16088.

[7] Lückerath D, Ullrich O, Speckenmeyer E. Modeling time table based tram traffic. *Simulation Notes Europe (SNE)*. 2012; 22(2): p 61-68.

[8] Middelkoop D, Bouwman M. SIMONE: Large Scale Train Network Simulations. In: Peters BA, Smith JA, Medeiros DJ, Rohrer MW, editors. *Proceedings of the 2001 Winter Simulation Conference*, Arlington; 2001.

[9] Nash A, Huerlimann D. Railroad Simulation Using OpenTrack. In: Allan J, Hill RJ, Brebbia CA, Sciutto G, Sone S, editors. Computers in Railways IX, WIT Press, Southampton; 2004; p 45-54.

[10] Schöbel A. A Model for the Delay Management Problem based on Mixed-Integer-Programming. In*: Proceedings of ATMOS; 2001.*

[11] Speckenmeyer E, Li N, Lückerath D, Ullrich O. Socio-Economic Objectives in Tram Scheduling. Technical Report, Universität zu Köln, to appear.

[12] Suhl L, Mellouli T. Managing and preventing delays in railway traffic by simulation and optimization. *Mathematical Methods on Optimization in Transportation Systems*; 2001. p 3–16.

[13] Transports de l'agglomération de Montpellier. Subra R, editor. Horaires tram 1. 2012.

[14] Transports de l'agglomération de Montpellier. Subra R, editor. Horaires tram 2. 2012.

[15] Transports de l'agglomération de Montpellier. Subra R, editor. Horaires tram 3. 2012.

[16] Transports de l'agglomération de Montpellier. Subra R, editor. Horaires tram 4. 2012.

[17] Transports de l'agglomération de Montpellier: Un réseau en étoile [Internet]. [cited 2013 February 21]. Available from: http://www.montpellieragglo.com/tam/page.php?id_rubrique=31

[18] Transports de l'agglomération de Montpellier: Le tracé du ligne 5 [Internet]. [cited 2013 February 21]. Available from: http://www.ligne5-montpellier-agglo.com/?page_id=16

[19] Transports de l'agglomération de Montpellier. Subra R, editor. Plan du réseau du centre de l'agglomération; 2012.

[20] Ullrich O. 2014. Modellbasierte Parallelisierung von Anwendungen zur Verkehrssimulation - Ein dynamischer und adaptiver Ansatz [dissertation]. Universität Köln.

[21] Ullrich O, Lückerath D, Franz S, Speckenmeyer E. Simulation and optimization of Cologne's tram schedule. *Simulation Notes Europe (SNE). 2012;* 22(2): p 69-76.

[22] Ullrich O, Lückerath D, Speckenmeyer E. Model-based parallelization of discrete traffic simulation models. Submitted to ASIM 2014 - 22nd Symposium Simulation Technique, Berlin; 2014.

[23] Verband Deutscher Verkehrsunternehmen e.V.: VDV-Standardschnittstelle Liniennetz/Fahrplan, VDV-Schriften 452; 2008.

An Agent-Based Approach to Increasing Space Utilization of Office Buildings

Štefan Emrich

Department of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria; *stefan.emrich@datengeschichten.at*

SNE Simulation Notes Europe SNE 25(1), 2015, 9-16
DOI: 10.11128/sne.25.tn.102 73
Received: April 10, 2014; Revised December 10, 2014;
Accepted: January 15, 2015;

Abstract. Based on a brief analysis of the status quo of office space utilization (section 1), a hybrid simulation model combining discrete event simulation (DES) and agent-based methons (AB) is developed in section 2. In section 3 it is to analyze some general characteristics of such office systems. Although academic, the results underline the huge potential benefit for an increased space utilization through utilization of mathematical simulation.

Introduction

In private businesses the cost factor is one of the main – if not the major – contributor to decission making. Nevertheless there is a certain blind spot when it comes to space related costs, which is partially induced by the status of owning representative buildings or a spacious office. But aside from this, the awareness for the potential savings (regardless whether of GHG emissions or financial ones) through an efficiency increase in space management is hardly existent, yet. Subsequently current approaches to reduce space related costs focus on buildings' operating costs instead.

1 Potential for improvement

The effect of this focus is illustrated in Fig. 1 and has been described by Zitter et al. Operating costs account for only 20% of the annual building-related costs (1^{st} bar in Fig. 1; left to right). They then provide benchmarks according to which roughly 40% (of the initial 20%) are capable of being influenced —- thus 8% of the total costs (2^{nd} bar). It is further possible to reduce the influenceable costs by 30% (= 2.4% of total costs; 3^{rd} bar). Assuming a realistic reduction of 50% in practice, the total costs can be reduced by a mere 1.2 percent (4^{th} bar)! It is thus obvious that this approach cannot contribute to substantial savings.

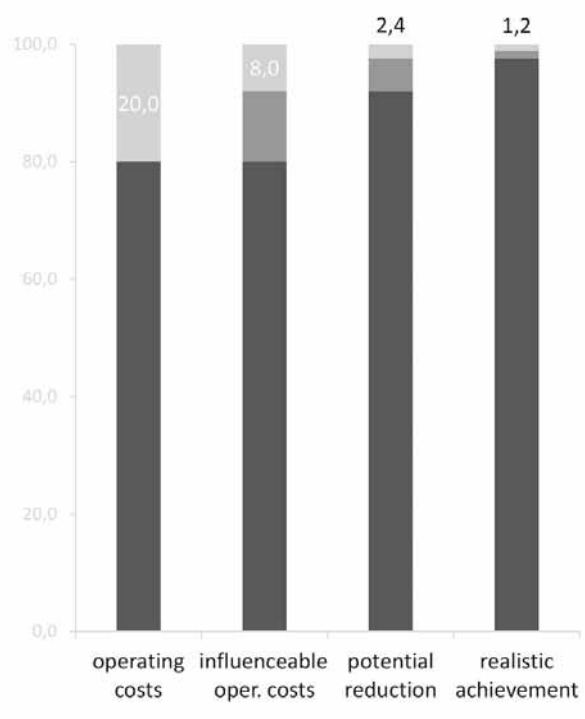


Figure 1: Practically achievable reduction of builing related costs by tweaking of operational costs.

On the other hand buildings are used only for a fraction of their life time only. As explained by Ottomann effective utilization of office space lies around a mere 5%. This includes a working week of five 8 hour work days, holidays and vacation of employees, breaks, sick leaves and social as well as organizational activities.

It is apparent that an efficiency increase in space utilization offers a far bigger potential for savings than reductions of operating costs. A theoretical increase of 5 percentage points (i.e. from 5% to a utilization of 10%) does equal cutting the space required in half – and thus reducing space-related costs by approximately 50%! This is illustrated by following example: A company with 100 offices has a utilization of 5%. Availability of hundred offices per week (7 days á 24 hours) is 16.800 office-hours ($100 \times 7 \times 24$). Utilization of 5% means that a mere 840 office-hours are actually "consumed" by the employees. As the actual need (840 office-hours) is not changed by a more efficient space-management, a raise to 10% utilization efficiency would require an availability of only 8.400 office-hours ($840 = 10\% \Rightarrow 100\% = 8.400$), which calls for (8.400 divided by 7 days á 24h) 50 offices – a 50% reduction of the original 100 offices.

The question that arises is: "How can (office) space be used in a more efficient manner?" - Which is equivalent to that of how much space is truly needed.

1.1 Static Approaches

As stated in Emrich et al., decision makers need to know how much space is truly needed to answer this questions. This of course is hard to answer without adequate information. Approximations can be derived by rule of thumb estimates customary in the particular trade, although they will remain (rough) estimates. Chances are that the need for space will be over- or underestimated. Both outcomes come with significant costs (see Kovacs et al. for financial insight on inefficient utilization of office buildings). Either there will not be enough space for all employees, which not only requires renting additional space, but also disrupts workflows and thus decreases overall productivity. Overestimating required work space, on the other hand, leads to sub-optimal utilization. The situation improves less than it could have.

Trying to improve the results, more detailed calculations could be carried out. Nevertheless these will become extremely complicated and complex when trying to incorporate different behavior of employees. For example will sales representatives have needs different from in-house account managers, who will again have needs, working- and vacation times that differ from those of the IT-staff. Getting exact results under such heterogeneous conditions is challenging, to say the least. In addition, even if it would be possible to obtain exact results for this problem, they would be valid only for this one scenario. A change within the employee structure or a different space management strategy would require starting from scratch, as all calculations and consideration have to be applied to the new scenario. Another flaw of this approach is that it neglects the stochastic nature of the observed system (i.e. employees are not robots that have ever repeating, non-changing routines within their work-cycles).

Another approach is to closely monitor and track the employees' actual work place needs and use the data obtained for statistical analyses (e.g. electronic monitoring of workplace activities, collecting information on employee position, etc.). Nevertheless, this approach has some major drawbacks. First it raises issues regarding privacy. And even if legal it is likely to cause bad blood among employees and/or staff associations. Second if monitoring systems are not installed yet, it is costly to do so. Further it takes a long period of time to acquire sufficient amounts of data. Third data gathered is, by definition, always historic – even in real-time systems. Thus it can only be used to explain and analyze (management) strategies, employee structures and office layouts that have existed and been monitored in the real world (i.e. those from which the data comes from). But the data is only of very limited use when trying to understand the effects of alternative scenarios (e.g. modified employee structure, different working times, changed space management, etc.).

These above two methods (rule of thumb and statistic analysis) are regarded static models as they do depict the system behavior, but without any change over time. This is not to be confused with an (in)ability to "predict" the future state of the system. But the prediction does not change over time, as well as the models itself do not change their states.

2 Model Implementation

To overcome the limitations of static approaches a dynamic model – based on discrete event simulation (DES) and agent-based methods – is being developed. The factors that need to be taken into account when modeling space utilization in an office environment are fairly similar to those of the described university environment. As explained in Emrich et al., these are:

- employee structure
  i.e. which employee types are within the system and how many employees of each type

- employee behavior
  i.e. which working preferences and what kind of behavior do the employees have (e.g. working times, fraction of field work, etc.)
- office environment
  i.e. how many offices/workplaces of what type are available
- The space management in place
  i.e. which rules have to be considered when it comes to assignment of work places, which workplaces are available for whom (employees or employee-types), etc. – these rules strongly depend on the objective of the simulation

2.1 Modeling Approach

In such a setup, as explained in Emrich et al., the individual employee can be regarded as the smallest unit. It is her behavior that defines the simulation result, and subsequently it is necessary to depict the employees in the most accurate way. For this reason "top-down" approaches (such as statistical methods) are only of limited success: they describe the system as a whole – without giving respect to the interactions of the system internal elements. Agent-based (AB) methods, as used in the present case for development of the "More-Space Office Tool", are producing the system's behavior via definition of its smallest units and their respective interactions — the employees and their behavior. AB modeling is treating every instance (i.e. employee) as an independent entity with an individual behavior.

Further, to recreate realistic behavior, the stochastic nature of events has to be incorporated into the simulation model. This is necessary as, for example, employees will not come to work every day at exactly 8:00 A.M. On the contrary they will most often come a bit earlier or later as they have to deal with "unexpected" events, such as traffic jams or delayed public transport. Such events can potentially trigger chain reactions (e.g. missing the first of a series of connections by only a second can lead to a cumulative delay of several hours) and are thus vital for the dynamic nature of the model. Discrete event simulation (DES) is aiming at such problems and is therefore incorporated into the model.

In order to combine the features of AB methods and DES a hybrid model was created using the simulation environment AnyLogic, which is based on the object oriented programming language JAVA and capable of supporting both approaches (AB and DES).

2.2 Employee

Within the model each employee is modeled as individual agent in a class called "Worker". This agent has several parameters and variables and a combined statechart for its health- and work-status (see Fig. 2). The parameters of the object Worker are:

- employeeType
  name of employee type (e.g. customer support or developer)
- employeeColor
  the color of the agent's visual representation
- daysInOffice
  the number of days/week which the agent is working in the office, (e.g. customer support might come into office only once a week and be at customers' locations the remaining days of the week)
- timeInOffice
  the average duration the agent stays in the office, once it comes to the office
- startShiftMin
  the earliest time that the agent will come to work (if it comes to work), i.e. earliest time to start its shift
- startShiftMax
  the latest time that the agent will come to work (if it comes to work), i.e. latest time to start its shift
- fixedWP
  boolean parameter if the agent owns a fixed (exclusive) workplace or not

And its variables are:

- assignedWP
  ID[1] of the workplace currently assigned; if fixedWP is *true* this ID is constant throughout simulation
- sickLeaveDays
  counter of days the agent was on sick leave
- spentVacation
  counter of vacation-days consumed
- sicknessDuration
  used to store the duration of the sickness if the agent turns sick

[1] Remark: pointer to object instance.

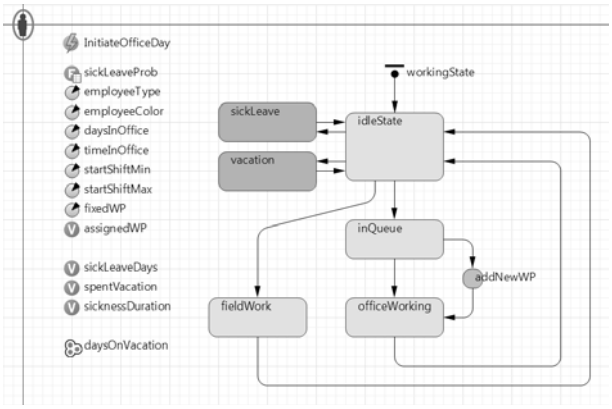


Figure 2: Structure of Worker object in the model, including parameters, variables, statechart and control-elements.

Besides the statechart and the above variables and parameters the Worker-agent is composed of three more control elements that are used to control the agent's behavior. These are *daysOnVacation* (list), *sickLeaveProb* (table function) and *InitiateOfficeDay* (dynamic event) and will be looked at in more detail later.

As mentioned the statechart used combines the working state and the health state of the agent. The reason for the combination is the assumption of their mutually exclusive nature. I.e. that an employee turning sick is not going to go to work. For simplification of the model it is further assumed that an employee is not turning sick during vacation or while at work. The top arrow indicates that the agent enters the statechart into state "idle" (*idleState*). This is the initial state from which every work day is started. From here the agent starts its working day either as an office day (via *inQueue* and *officeWorking*) or working outside the office (*fieldWork*) before it returns to the idle state. In case that the agent turns sick or takes a day off it changes from idle into state *sickLeave* and *vacation* respectively. The last remaining state, *addNewWP*, will be described later.

At initialization of the simulation each agent's *daysOnVacation*-list is filled with 25 days[2] on which the agent is on vacation. These 25 dates are scheduled randomly with following constraints:

- For 50% of all employees a blocked vacation is being scheduled, ...
  - with a length that is uniformly distributed between 10 and 15 days, and
  - which starts randomly (uniformly distributed) within a user-specified "core vacation period" (e.g. summer holidays).
- For 40% of the remaining employees (20% of total) a blocked vacation (of uniformly distributed length between 10 and 15 days) is scheduled at a random time (unif. distrib.) during the year.
- Finally each employees vacation list is filled up (until 25 days are reached) with random (unif. distrib.) days.

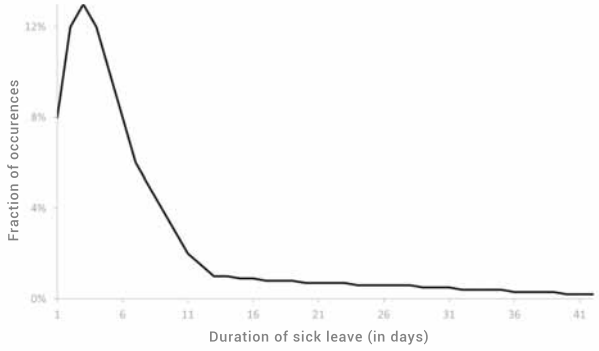


Figure 3: Skewed bell-shaped distribution of sick-leave duration used within the office model: length of sick leave duration (x-axis) as fraction of total occurences (y-axis) of sick leave.

To create a closer to reality behavior of employees a sickness function has been developed. Based upon several data sets from "Statistik Austria" (the Austrian Statistical Central Office) a distribution of the duration of sick leaves has been developed (see Fig. 3). This distribution is naturally only a rough estimate, but sickness duration and frequency strongly depends on the business field and the region/country and can thus not be modeled precise and generally valid at the same time. Instead it is recommended to use real-world data for parametrization and model fitting.

2.3 Employee Structure

With the basic employee being defined in the above described, flexible way, one object class (Worker) can be

[2] By Austrian law employees are entitled to an annual vacation of five weeks. Hence people working five days per week (the majority) receive 25 days of vacation (per year). People working 6 days per week subsequently receive 30 days.

used to represent different employee types. At initialization of a simulation run the user is presented with a GUI (see Fig. 4) that allows to define the numbers of employees per employee type, the core vacation period as well as choosing between flexible and fixed (i.e. individual, not shared) workplaces.

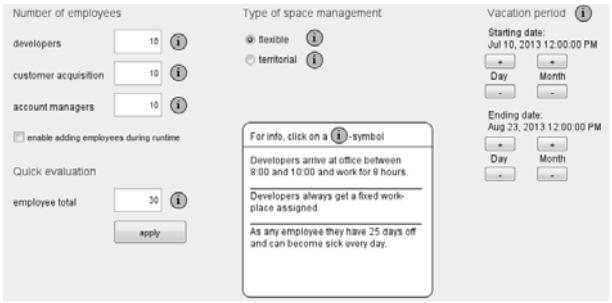


Figure 4: Section of GUI that allows specification of main simulation-parameters.

The model then generates the specified number of employees per employee type (in this case three types are hard-coded) together with their specific characteristics (see parameters, listed before).

An alternative possibility is to control the employee structure using spreadsheet-files, which are loaded by the model at initialization. This approach maximizes flexibility and empowers users without programming skills or access to the source code to precisely control the employee structure. In this case the model processes the spreadsheet-file row by row and creates an employee type (with multiple instances) for each of these. The exemplary spreadsheet depicted in Fig. 5 leads to the creation of three different employee types: "Developers", "Acquisition" and "Forenoon", with respective behavior.

	A	B	C	D	E	F	G	H	I
1	Employee statistics								
2									
3				arrival at office			agent color		
4	type	No. of employees	days in office	earliest	latest	time in office	red	green	blue
5			days/week	hour	hour	hours	0...255	0...255	0...255
6	Developers	30	5	8,00	10,00	7,00	255	165	0
7	Acquisition	30	1	8,00	10,00	2,00	245	20	147
8	Forenoon	40	5	7,00	17,00	2,00	84	84	84
9									

Figure 5: Spreadsheet file controlling model-internal employee structure.

With this approach it is even possible to create an individual class for every single employee and thus incorporate individual behavior (e.g. by adding preferred vacation times and individual, age- or gender-dependent illness probabilities). Besides potential privacy concerns this naturally requires to have the corresponding data in the first place. In relation to the benefit for the simulation result this approach is most likely too costly and thus not reasonable to pursue.

2.4 System behavior

In the current implementation the general goal is to use the model to calculate the number of required workplaces – for a given employee structure – and in this particular case to evaluate the savings potential compared to fixed workplaces[3]. With the employee structure and behavior in place, the next step is to model the general system behavior.

At initiation the model creates all employees as instances of class Worker, which, when coming into office require a workplace. Subsequently the number of required workplaces can be obtained by adding a workplace to the (virtual) building each time one is needed. This approach is supported by the object oriented architecture of the simulation environment. It is possible to create a new instance of a class at runtime. Thus offices and workplaces are implemented as classes (*Office1WP* office with one workplace, *Office2WP* with two workplaces and *Workplace*). When a *Worker* changes its state from idle to *inQueue* a check for free workplaces is performed. If a workplace is available the employee uses it, if not a routine is called which creates an additional instance of *Workplace* and assigns it to the employee. Remark: Since employees with *fixedWP* = *true* do not release their workplace (ID), it also cannot be taken by other employees.

The structure of class *Workplace* is a fairly simple one. Besides information relevant for visualization purposes (e.g. its coordinates) it has following three variables:

- assignedEmployee
  analogous to *Worker*'s *assignedWP* this holds the ID of the employee that is assigned to the workplace; in case of a fixed workplace the ID[4] does not change

- nTimesUsed
  counter for utilization analysis which registers ever use-session

[3] I.e. a workplace model in which every employee has an individual workplace that is not shared

[4] Remark: pointer to object instance.

- nMinUsed
  counter for utilization analysis which registers every minute of usage

As indicated the model has two more object classes: *Office1WP* and *Office2WP*. They represent the frame in which workplaces are set. Depending on the 1WP or 2WP the object houses one or two workplaces. The primary (and sole) reason for their existence is the visualization of the simulation.

During simulation a cyclic event *wakeAgents*, scheduled for 00:01 of each simulated day, triggers the actions of all *Worker*-instances. It schedules their dynamic events *InitiateOfficeDay*, depending on probabilities and other factors. First it checks whether the current day is a workday or not. If it is, it checks how many days per week the respective agent is working "in office" and probabilistically determines whether a day in office or a day of field work is scheduled.

The exact point in time when the (employee-internal) dynamic event *InitiateOfficeDay* is taking place is scheduled with a uniform distribution between the agent's *startShiftMin* and *startShiftMax*. When this point of time is reached, this internal dynamic event performs a state-check on the agent (if healthy and not on vacation) and then (probabilistically) determines whether the agent becomes ill or not — in which case it proceeds to work in office.

3 Simulation and Findings

3.1 Parametrization

Without real-world data to derive an employee structure from and with no benefit of a super-realistic one, a simplified employee structure was used to evaluate the savings potential of a flexible space management compared to a fixed workplace model. Nevertheless expert-knowledge was used to obtain a close-to-real employee structure and behavior.

The structure consists of developers, employees in customer acquisition and account managers, of which only the first are granted a fixed workplace. This is explained by their respective "in office" working times. Developers come to office (if not ill or on vacation) on all workdays, arrive in office between 8:00 and 10:00 (a.m.) and work for 8 hours. Customer acquisition personnel spend four days per week with field work and hence only have one day in office. When they are in office they, as developers, come between 8:00 and 10:00 and work for 8 hours. Account managers are doing mainly field work, but come into office daily, although at irregular and changing times. They arrive between 8:00 and 17:00 and are then in office for two hours.

3.2 Results

For evaluation of the savings effect through flexible workplace utilization a balanced structure with $1/3$ of every employee type was used. Flexible workplace utilization was defined in such a way that all employees (except developers) use any free workplace (except those that are assigned to a developer); developers always use their assigned workplace.

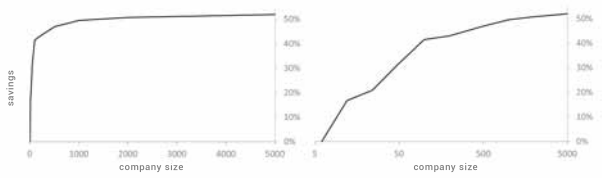


Figure 6: Workplace savings potential (y-axis) as a function of company size (x-axis) – plotted on a linear (left) and a logarithmic scale (right).

The model was used to simulate the workplace requirements of each company size over one year (365 days). To compensate the effect of outliers the Monte Carlo method was applied. For every company size 10 simulation runs (each with a random seed for random number generation) were produced and averaged. The savings potential is then calculated as the difference between the simulation average and the company size[5]. The results are shown in Fig. 6, once on a linear (in figure left) and once on a logarithmic scale (in figure right). It is obvious that small enterprises can draw no and medium-sized ones only limited benefit of a flexible use of workplaces. Large companies on the other hand can cut more than 50% of their workplaces, compared to a fixed (workplace utilization) model!

Arguably averaging of simulation results leads to a lower number of workplaces than required in the "worst case" scenario. But then again flexible use of workplaces always holds a theoretic danger of shortage: in the absolute "worst case" all employees require a workplace at the same time. The question that has to be answered in practice is: how much risk does one want to

[5] As the number of employees equals the number of workplaces required if every employee has their individual workplace.

take? Depending on the answer it is necessary to plan with a sufficiently large buffer. In addition the difference between mean and maximum is very small (see tab. 3.2), which is explained by the (fairly long) runtime of the basic simulation (365 days). The odd employee numbers are explained by the employee structure, which consists of three equally large groups.

Company size	average	maximum	difference
6	6	6	0
12	10	10	0
24	18	19	1
28	32	33	1
99	58	58	0
198	110	113	3
501	260	266	6
1002	501	506	5
2004	981	988	7
5001	2395	2405	10

Table 1: Simulation results for required workplaces: average and maximum (of 10 simulation runs) and difference.

Using the variable *nMinUsed* of the *Workplace*-object it is possible to calculate the effective occupancy — either for every workplace, or for the whole lot. In doing so one must consider that the employees are modeled in such a way that they do not leave their workplace until they finish their workday. I.e. there are no meetings, conferences, lunch-breaks, and the like — which would naturally reduce effective occupancy. Incorporation of such elements would require to consider whether an unoccupied workplace left for such a reason would become available (for use by another employee) or remain reserved although unoccupied by the initial employee. The present implementation has been chosen in order to avoid this problem.

Occupancy has been calculated in two different ways. Once the total time of workplace-usage is divided by the total simulation time (i.e. 24 hours, 7 days a week), the second time it is seen as fraction of the core time (10 hours per business day, i.e. Monday through Friday). With respect to the findings in Fig. 6 a company with 500 employees can already profit significantly of a flexible workplace management. Thus occupancy has been analyzed for this category by simulating a period of one year for 5 times with the flexible workplace utilization as previously described. I.e. developers, who are working full time, have a fixed place, the remaining employees not.

The simulation results (depicted in Table 3.2, labeled “fixed”) show that the total occupancy (labeled “total”) lies at about 19.7% and during core time (labeled “core”) around 66.3%. In a second step the flexibility of the space management has been increased by one notch: developers also use flexible workplaces (results labeled “flexible” in Table 3.2). Even though developers are working full time (i.e. 8 hours per day, 5 days per week) the impact that this change has is dramatic. Only by unblocking the (previously fixed) workplaces blocked during vacation and illness of developers total utilization was increased by one percentage point (relative: 5%) and core utilization by 3.3 percentage points (relative: 5%). Again (compare to tab. 3.2) the deviation of the results is so small that the ranges of the two settings’ results never overlap. The explanation for this is again found in the long simulation period of 365 days, which causes graduation of a lot of random influences.

	total (%)		core (%)	
sim-run	fixed	flexible	fixed	flexible
1	19.88	20.87	66.87	70.19
2	19.51	20.11	65.61	67.66
3	19.63	20.87	66.02	70.21
4	20.03	21.14	67.38	71.10
5	19.29	20.48	64.89	68.87
average	19.70	20.69	66.28	69.61

Table 2: Comparison of workplace occupancy for scenarios “fixed” and “flexible”; simulation results for “total” occupancy and “core” time occupancy (10 hours/business day).

3.3 Conclusion

Calculation of required number of workplaces is a queuing theory problem such as establishing the number of checkout counters in a post office or supermarket and general dimensioning problems in the area of service provision. With the above described hybrid ap-

proach (combining AB methods and DES) it is possible to model workplace utilization at an employee-based level, which has several advantages.

One major advantage is the transparency of the model and hence of the simulation results. It allows the user for whom simulation is carried out to understand and follow the reasoning behind the model, without becoming a simulation-expert herself (e.g. statecharts are easily read and understood). A second is the closeness of the attribute-mapping between implementation and reality, which at the same time is partially responsible for the before mentioned. An employee turning ill is implemented as agent-internal state change from "healthy" to "ill".

Besides the easy interpretation of the implemented model, the object oriented approach also allows for a very flexible and easy modification and adaption. If additional requirements arise they can more often than not be incorporated in a very efficient way (e.g. fine-tuning of agents' behavior, adding of attributes to objects, introduction of additional statistics, etc.).

With the model being of academic nature, the focus of it lies on serving as a proof of concept. These results obtained are thus not carved in stone, as they are a product of employee structure and behavior as well as of the space management in place and of simulation parametrization. All of which was based not on real data but on assumptions, which, although chosen with a claim for authenticity, reflect the simplifications accepted to obtain a slender model.

Against the background of current practice in space management of office buildings the model results point at a huge potential for improvement. Exploitation of it requires – amongst others – raising awareness for the issue, which can be supported by such conceptual models. On the other hand the model already incorporates most of the aspects necessary to conduct analysis of real systems and only calls for appropriate parametrization. In case of potential extensions, the flexible (object oriented) architecture allows for very efficient adaption. Finally, the big question and challenge that remains is whether an institution is willing and capable of incorporating the required changes of business processes, i.e. installing a flexible space management. Without this step the potential for improvement is, as pointed out intially, more than limited.

References

[1] Ottomann H. Wussten Sie, dass Büroflächen nur zu fünf Prozent genutzt werden? Es ist Zeit! *Mensch & Büro*. 1994;2:28–30.

[2] Emrich Š, Wiegand D, Kovacs A. A Mathematical Simulation Tool for Increased Space Utilization Efficiency. In: *Proceedings of EFMC 2013 (Prague, May 2013)*. 2013; .

[3] Kovacs A, Štefan Emrich, Wiegand D. Strategies and Methods to improve Space-Utilisation within Companies. In: *Proceedings of EFMC 2013 (Prague, May 2013)*. 2013; .

Simulation System for Radiology Education Integration of Physical and Virtual Realities: Overview and Software Considerations

Ali A. Alghamdi

Department of Radiological Sciences, College of Applied Medical Sciences, University of Dammam, P.O. Box 2435, Dammam - 31451, Saudi Arabia; *med.alghamdi@gmail.com*

Simulation Notes Europe SNE 25(1), 2015, 17 - 26
DOI: 10.11128/sne.25.en.10275
Received: February 20, 2015; Revised April 20, 2015;
Accepted: April 20, 2015;

Abstract. The aim of the proposed system is to give the students a flexible, realistic, and interactive learning environment to study the physical limit of different postures and various imaging procedures. The suggested system will also familiarize the students with various imaging modalities, the anatomical structures that are observable under different X-ray tube settings and the quality of the resulting image. Current teaching practice for radiological sciences student asks students to simulate the imaging procedure in role plays – a student to be a patient and the other as the radiologist. Other ways include the use of physical phantom with bone and soft tissue equivalent material but still the use of X-ray have to be used with all the requirement of such examination to be in place, e.g., room shielding, lead apron, and other radiation protection procedure. The proposed system has several physical components and virtual components. Students manipulate the mannequin into the model of the imaging modality and in a posture suitable for the purpose of the imaging study. The virtual components of our simulation system include a posture interface, a computational phantom generator, and a physics simulator. The synthetic image will be produced and conform to the Digital Imaging and Communications in Medicine standard so that it can be stored, retrieved, and displayed in a standard picture archiving and communication system that hospitals use.

Introduction

Simulation nowadays covers a wide spectrum in meaning and in application depending on the field that it is applied. Among all fields of sciences, the flight simulator for training pilots is perhaps the best-known simulation system. It familiarizes the trainee with the airplane cockpit and different situations that may arise. Before the trainee pilot takes off in a real airplane, he has already accumulated valuable experience of difficult situations that may not arise even throughout his career as a qualified pilot, and without risk to himself and passengers. The idea had been taken-up well by educators in many disciplines and extended to medical sciences in which the well-being of patients is of paramount importance. Simulations alleviate the needs to practice on real patients to sharpen the skill of the clinician.[1] In fact, simulation was said to be the 'ethical imperative' in some areas of medical education.[2,3] Furthermore, simulations were shown to offer improved learning and better knowledge acquisition and retention when compared to conventional lectures[4] because skills were learned and reinforced in an iterative manner.[5] They are and will be an important aspect in all medical training.[6-9] Within medical sciences and related radiology training, simulation systems have been developed for, for examples, surgery[6,8,10] cardiac examination,[11] catheter administration,[12] ultrasound examination[13,14] to name but a few.

In addition to the balance between patient safety and clinical skills, radiological education also poses a further risk of radiation exposure to the students if practiced on real equipment that utilizes ionizing radiation such as X-ray and computed tomography (CT).

Simulations remove this concern completely from the learning process. In the education and training of radiographers, radiologists, and other related areas of radiology, the trainees study various X-ray-based imaging systems and the procedures that imaging examinations are conducted. The training requires their understanding of the positioning of the patient in the imaging system and what they expect to be observable in the resultant images. Some teaching practice asks students to simulate the imaging procedure in role plays – a student to be a patient and the other as the radiologist. The major drawback in the exercise is in the fact that an actual radiographic image cannot be obtained due to radiation risk. The learning experience is also limited by the availability of relevant images. Computer simulations overcome many inconveniences in role plays. The two teaching methods are not mutually exclusive but complementing each other. The values of simulation in the medical education have been argued favorably in various disciplines,[7,15-18] and radiology.[19,20] Radiological simulation packages have been developed for the purpose.[20-23]

In general, there are several technologies commonly in use for simulations: motion tracking, mannequins, image libraries, and synthetic images. Our aim is to give the students a flexible, realistic, and interactive learning environment without exposure to radiation. We are proposing a new concept in radiology simulation that combines physical and virtual reality. For example, if a student is taking an image of the wrist a patient, we want him to "take images" of the hand in different orientations in the imaging system. Different features of the wrist are observable with these orientations. Therefore, there are two aspects of the simulation – physical positioning of the patient and the X-ray images obtained from the position. The following is a short description of the candidate technologies for the purpose of mixed reality simulation – motion tracking of markers and mannequins for patient positioning; image libraries and synthetic radiography for image generation.

1 Motion Tracking

Motion tracking has been used successfully in movie production and computer game developments.[24,25] Markers are placed at the joints of an actor, and his motion is captured by the camera.

Placing the markers on the student can provide the positioning data to a computer for radiographic image generation. The advantage in this technology is that the students will develop the empathy for the patient because they personally tried out the position. The drawback is in the system setup that will require time and great care. This may be impractical even for a small class of students.

2 Mannequins

Besides automobile safety studies, mannequins have also been used extensively in medical education simulations, for examples, in obstetric trainings,[26] the Harvey mannequin in cardiology training,[27] the Gas Man[28] and the Comprehensive Anesthesia Simulation Environment[29] in anesthesia. Interested readers may find excellent reviews of mannequin applications in medical training in references.[26,30] Since we want a realistic simulation, we need a life-size dummy with flexible joints that are similar to human joints. Car safety studies have been using such dummies for many years.[31]

For our purpose, the dummy should have a realistic human form and features including some soft tissues. Realistic internal structures are not needed, but a few anatomical landmarks that can be felt on the skin, such as the base of the skull and the tip of the pelvis, are essential. Ideally, the dummy must be light enough for manipulation by one or two people without lifting devices. Sensors inside the dummy will supply the position and the orientation of all the joints to a connected computer. From these data, the computer will 'generate' the corresponding X-ray images.

3 Synthetic Radiography

The second approach is to have a computational human model whose posture is made according to the dummy's data. Then, a simulation of the X-ray through the computational model generates the radiographic image. The technology required in the computational model approach is tangible. Accurate computational human models with organs and tissues are have been used in radiological studies for many years.[32-36] The posture of all these models is limited to the supine position that is the position of the CT and magnetic resonance imaging (MRI) scans.

Unlike animations in movies and computer games that require external features of a character but not the anatomically correct structures inside, a computational model for radiological imaging demands anatomical details. Such details move and deform with the posture. Recent developments have shown possible techniques.[37] Furthermore, new modeling techniques and increased computing power allow the generation of simulated images from computational phantoms directly.[38] Along the same line of study are the simulations of CT, positron emission tomography (PET) and single photon emission CT scans. The mannequin will be lying on the couch that can extend through the borehole of a mock-up scanner. In these cases, the position of the couch is also required.

4 System Design

Figure 1 is a schematic drawing of the system. The mannequin [part 1 of Figure 1] has realistic external shape and flexible joints with rotation/position sensors.

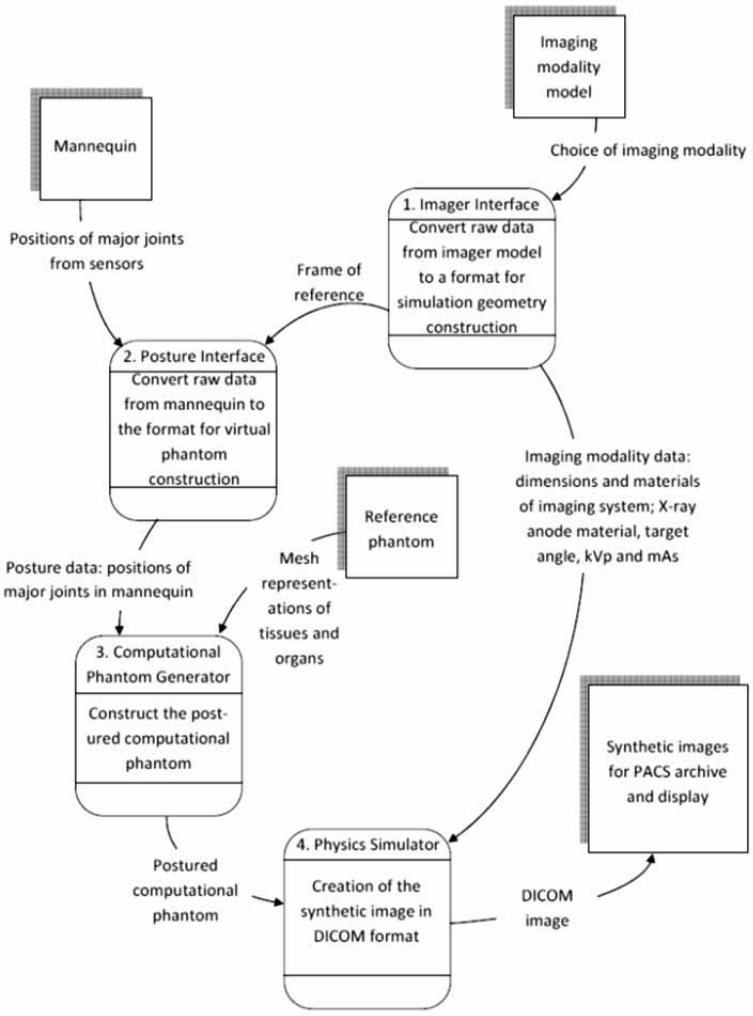


Figure 2: Data flow diagram of the proposed interactive simulation system. The reference phantom is generated only once for the corresponding mannequin, but it starting point for creating the postured phantom.

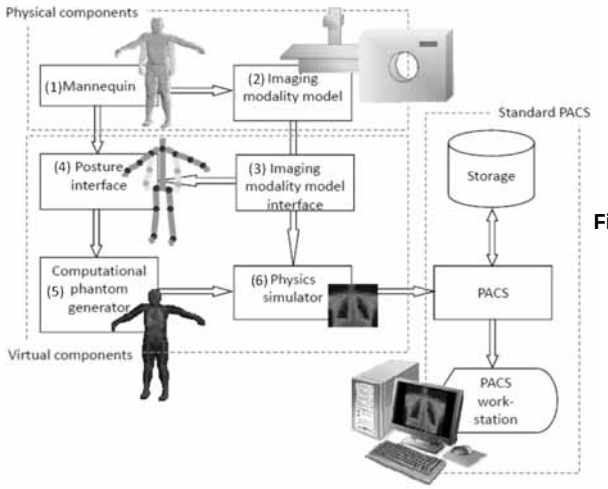


Figure 1: Schematic drawing of the proposed interactive simulation system. The mannequin and the imaging modality model are the physical components of the system. The data from these components are fed into the virtual parts that carry out the simulations with a computational phantom. The simulated images can be stored, displayed, and analyzed with picture archiving and communication system.

The angular/positional data are fed into the posture interface [part 2 of Figure 1] that tracks and displays the mannequin's posture. When the mannequin is ready for imaging, the computational phantom generator [part 3 of Figure 1] builds the phantom according to the posture. This computational phantom will be available to the physics simulator [part 5 of Figure 1]. Furthermore, the mannequin is positioned in an imaging modality model [part 4 of Figure 1]. The mannequin's position relative to the imager and the imager settings is also supplied to the physics simulator. The imager's settings include X-ray generator and detector characteristics. The physics simulator constructs the simulation geometry and tracks the X-ray photons from generation in the X-ray generator to their absorption in the detector. In the case of three-dimensional modality simulations, volumetric images are also reconstructed.

The images will conform to the Digital Imaging and Communications in Medicine (DICOM) standard so that integration with a fully functional picture archiving and communication system (PACS) is possible. The DICOM standard and the PACS systems are used extensively in a clinical environment.

In the case of simple X-ray imaging of the extremities, partially constructed mannequin corresponding to the extremity in question can be used. In the case of PET simulations, the image modality model will not supply the X-ray tube details. It will allow the user to enter the pharmaceutical information. This information will be available to the physics simulation. The use of the simulation system is described with the data flow diagram illustrated in Figure 2.

In the cases of PET simulations, the description of the source starts with the distribution of the radionuclides in the body instead of the description of the X-ray source. However, the manipulation of the mannequin, the generation of the postured computational phantom, and the physics simulation remain essentially the same as CT and other X-ray imaging simulations.

4.1 Mannequin

The mannequin [part 1 of Figure 1] is composed of a light-weight aluminum skeleton structure and a polysilicon skin, giving a realistic external shape of a person. Anatomical landmarks are attached to the aluminum skeleton. They can be felt by the user through the soft polysilicon skin.

The skeleton structure is connected by multiple-axis joints with rotation sensors and/or radio frequency (RF) transmitter. The locations of these joints correspond to the following joints in a human skeleton – the cervix between the base of the skull and the spinal column, shoulders, elbows, wrists, hip joints, knees, and ankles. The rotation sensors are connected to the computer system via cables or wireless system. The mannequin is manipulated by the user into the imaging modality model and in a posture suitable for the modality. The angular information of each sensor is fed to the posture interface. Position data received from the RF transmitters also allow the posture interface to deduce the mannequin's position in the imaging modality model.

4.2 Imaging modality model interface

The imaging modality model interface [part 2 and 3 of Figure 1] accepts input of the imaging parameters from the user. The parameters include filtration, tube voltage, and tube current of the X-ray generator in the case of simulating the X-ray-based imaging modalities or type of radiopharmaceutical and its concentration in the case of simulating nuclear medicine procedures. These data are supplied to the physics simulator for generation of virtual radiation particles. The control of these functions can be incorporated into the graphic user interface (GUI). The GUI also can reflect specific look and shape that of standard normal digital X-ray or other modalities accordingly. The GUI thus familiarizes the student for the GUI system on the real modality in a clinical environment.

Posture interface. The posture interface [part 4 of Figure 1] reads in the data from the rotation sensors and/or RF position system. A stylized visual representation of the mannequin and the imaging modality model are displayed on the screen for the benefit of the user. When the user is satisfied with the simulation configuration that includes the mannequin posture data and the imaging parameters, the posture interface will forward the data to the computational phantom generator and the physics simulator.

Computational phantom generator. The function of the computational phantom generator [part 5 of Figure 1] is, as its name implies, to construct a computational phantom. The construction starts with the data from the posture interface which provides the information of the selected joints in the mannequin.

This is also the information on the corresponding joints in the computational phantom. Thus, the computational phantom generator has a built-in reference phantom. This reference phantom is created from CT/MRI scans of a real patient in a supine position [Figure 3].

The three-dimensional image is segmented so that tissues and organs are individually identified. The mannequin's dimensions and joint positions are derived from this reference phantom so that the selected joints in mannequin reflect those in the reference phantom. Furthermore, the morphology of mannequin skin is also derived from the reference phantom.

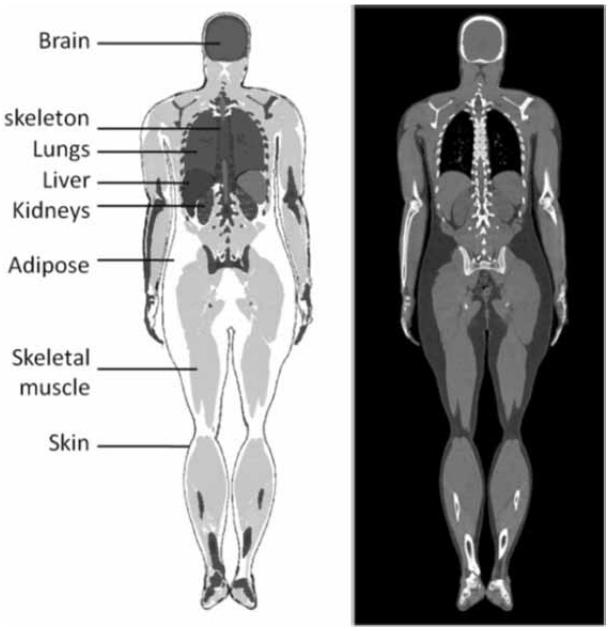


Figure 3: A voxel phantom (left) created from a computed tomography image (right). Each tissue is labeled by a number (tissue ID) after segmentation. In this coronal cross section of the phantom, the tissue IDs are plotted in different colors. There are about 100 identified tissues or organs in the phantom; only a few of them are shown here for illustration.

Creation of the built-in reference phantom. The reference phantom is created by segmenting the CT/MRI images. Triangular meshes [Figure 4] are generated to delineate the major organ and tissue outlines, including bones and skin as well.

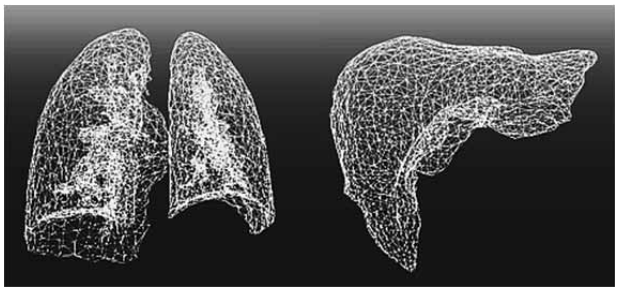


Figure 4: Mesh representation of the lungs (left) and the liver (right).

This constitutes the set of complete morphological data. Then, a reduced morphological data set is constructed. This reduced data set consists of the meshes of the bones and the external skin outline.

A process known as skinning and rigging [39] is applied to the bone and skin meshes such that the skin mesh is attached directly to the bone meshes. The process is a computer graphic technique used extensively in game and movie industries. In cases of suitable patient scan is not available, another method for the phantom generation can be used, e.g., NURBS-based.[40]

Generation of the posture phantom. Starting with the reduced morphological data and from the joint positions and orientations, the computational phantom generator maps the skeleton to the posture of the mannequin. Since the skin is attached to the skeleton directly in this data set, the skin mesh of the reference phantom is deformed accordingly. This creates a simplified posture phantom with bones and skin only. Then the deformation of internal organs and tissues in the complete morphological data is interpolated from this simplified phantom to obtain the detailed posture phantom. The last step is to voxelize the detailed posture phantom and to associate each voxel with elemental composition and density of the tissue of the voxel.[41] The elemental compositions and densities are coming from literature. This voxelized phantom is the computational phantom required in the subsequent physics simulations.

Major software graphic user interface modules. In addition to the basic function of any given GUI here, it can play a role in aiding the education process when using such simulation system. GUI modules can introduce the student to the theoretical part of the imaging process. The GUI functions are to control the workflow of the simulation system, invoke the posture interface, imaging modality GUI interface and physics simulator GUI interface. Imaging modality GUIs interfaces accept the imaging parameters such as kilovotage peak (kVp)

and (mAs) in X-ray-based modalities. In the case of radionuclide distribution modality like in PET, the GUI provides the control of another related parameter such as detector material and pixel dimensions from the user. It is possible that certain information like the distance between the X-ray source and the detector can be obtained automatically from the imaging modality model. In the initial development stages, we might want this to be entered through the GUI.

5 Physics Simulator

The physics simulator [part 6 of Figure 1] uses the data from the imaging modality model to create the appropriate models of the radiation source and detectors. It generates and tracks virtual radiation particles through the imager and computational phantom geometry.

The particle generation depends on the radiation source. In the case of simulating X-ray-based imaging modalities such as CT, chest X-ray, dental X-ray, knee X-ray, and others, the anode-filter combinations are taken into account. The user chooses the tube potential (kVp) and current (mAs). In the case of nuclear medicine-based imaging modalities, the type of radioisotope, its activity, and distribution in the body are specified by the user. Thus, the physics simulator is generating the virtual photons at energies, positions, and directions relevant to the study.

The simulator then tracks the photons through the geometry using a combination of Monte Carlo and deterministic techniques. In the process, the radiation physics relevant to diagnostic imaging are taken into account. They include photoelectric absorption, Rayleigh scattering, and Compton scattering. Figure 5 shows a typical Monte Carlo process.

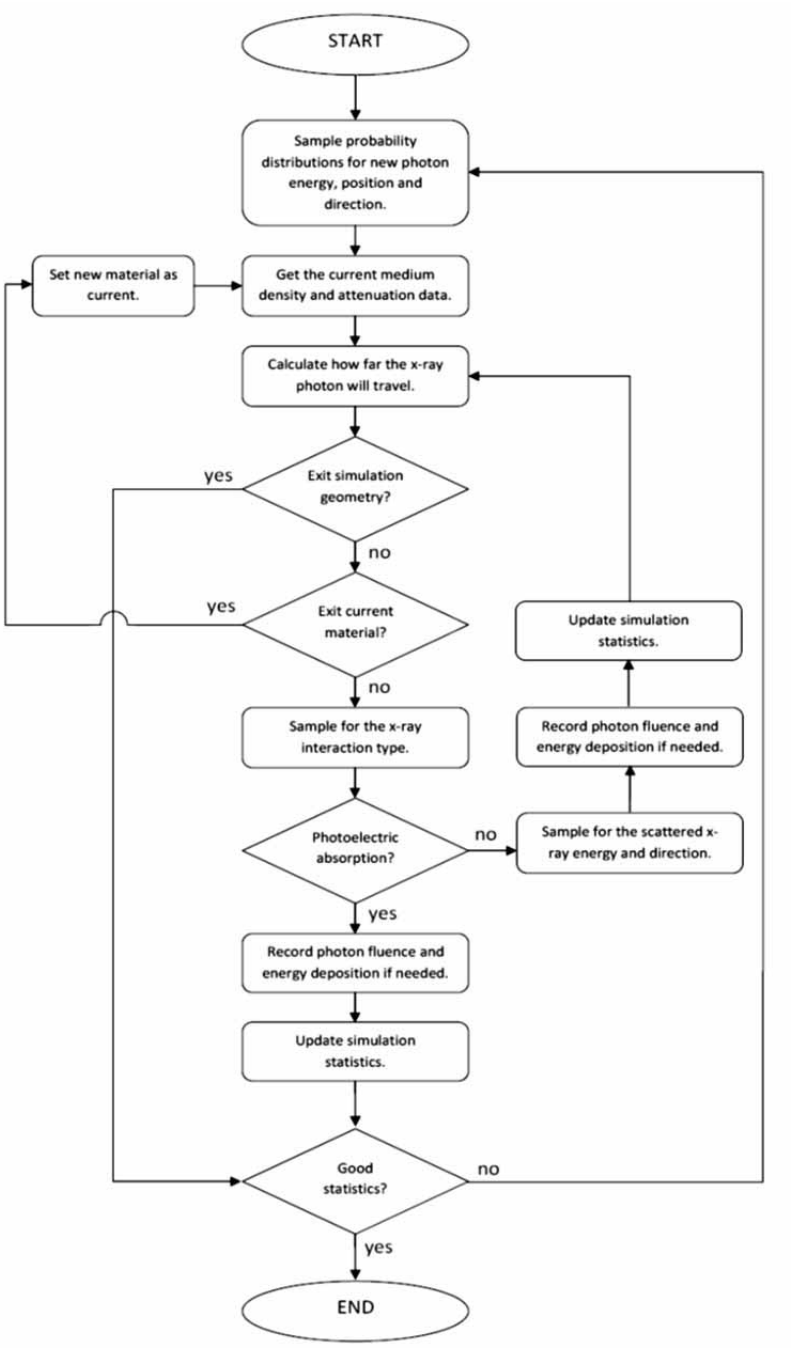


Figure 5: The Monte Carlo process to track one virtual X-ray photon. The process is repeated as many times as necessary to synthesize the image. It is important to note that the virtual photons must be independent of each other, that is, the sequence of random number in the sampling of the probability distributions cannot repeat itself in the simulation.

For a virtual photon at the source location, the simulation engine computes how far the photon will travel before interaction. Then it will determine the type of interaction and its outcome by sampling appropriate probability distributions from physics.

Each time the X-ray photon crosses from one material to another, the simulation engine will fetch the correct probability distributions for the new material and start tracking by computing how far from the boundary that the photon will go. If the photon enters the imaging detector, a score will be registered the process is continued until a sufficient number of photon satisfy the statistical benchmark of slandered Monte Carlo code scoring criteria.

Since the image is generated from the scoring of the virtual photons arriving at the imager, realistic images can be created so that students can observe the effect of patient posture and their choice of radiation source and imager. Each image is synthesized under a unique setting. Observable features differ in each image. Students can thus familiarize themselves with the choice of anode-filter combination, kVp and mAs settings in X-ray-based imaging and radioisotopes and their activities in the cases of nuclear medicine. Figure 6 contains several synthetic X-ray images of the reference computational phantom in [Figure 3] different X-ray energies and orientation. Volumetric images (for example, CT and PET) are also reconstructed from the scores. All images are converted into DICOM format for storage and display by PACS.

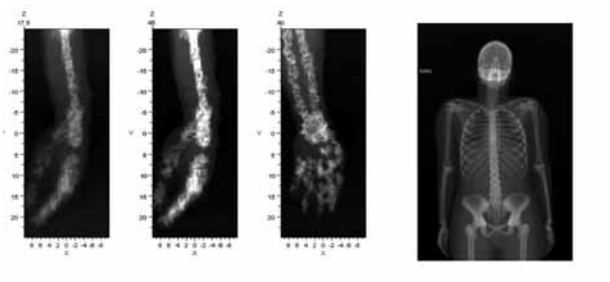


Figure 6: Synthetic X-ray images of the wrist of a low-resolution voxel phantom [36] and upper torso from the voxel phantom shown in Figure 3.

6 Discussion

In medical education, there is a concept of standard patient who is a person trained to pretend to be a patient. The standard patients provide uniform training to and unbiased assessment of the trainee physicians. In fact, simulation is becoming an integral part of the accreditation process in several medical disciplines.[15,17,42]

Our proposed interactive simulation system offers the radiological science trainees as the standard patient to trainee physicians. Although the mannequin would not be able to position itself automatically like a real person, abnormalities could be embedded in the computational phantom. Synthetic images with these abnormalities are possible.

The ability of the simulating abnormalities depends on the available computational power and resolution of the computational phantom. As an example, small fractures in bones might be difficult to model and impossible to simulate with a phantom of large voxels. On the other hand, small fractures can be simulated with a stylized mathematical model. Developing a phantom that support the simulation of small features will be a challenge. Techniques have been developed to address the challenge in the simulations of tumors and microcalcifications in the breast.[43-45] Such techniques are transferable to other radiological abnormalities. Further progress will be made in this aspect in the next stage of the simulation system development.

Radiation risk is a major concern in the career of a radiological technologist or scientist. This is especially true during training. The students might be preoccupied by the unfamiliar equipment and therefore paying insufficient attention to radiation safety. The proposed system attacks this issue on two fronts. First of all, there is no radiation involved physically. All types of radiation are occurring in the virtual world. The students are not exposed to the radiation. Any mistakes will not incur radiation exposure to the students. Second, radiation dose calculation can be carried out in real time as the radiation particles are tracked in the Monte Carlo simulations as the synthetic images are generated.

Without worries of radiation, the students will be able to familiarize themselves with the equipment at their own pace and in their own time. Effects of patient position and settings of the imaging equipment on the resultant image are available to the students immediately. Results can be archived for further analysis.[20] Student performance can be monitored online. Minimal supervision will be required from the educator. Clinics and hospitals can make better and more effective use of the resources for the benefit of the patients. These invaluable resources include clinicians and equipment.

When the students enter the real clinical environment for the 1st time, they are expected to have mastered the basic skills with respect to the equipment and radiation safety. They are also expected to understand the positioning aspects when dealing with a real patient. They have mastered those skills without any radiation exposure and without using up valuable clinical resources. Another advantage of the system comes from the versatility of the Monte Carlo method. Although PET and X-ray procedures are very different equipment-wise and clinically, the codes to track the radiation particles are the same. Thus, the simulation system can easily be expanded to include various radiological trainings in CT, nuclear medicine, radiotherapy, and others. More technical consideration can be found in.[46]

7 Conclusion

The idea and the design of an interactive simulation in radiological education have been presented, highlighting the software aspects of the system. This is an ongoing project where the system itself currently under development and prototyping. The hardware integration will be presented in a separate paper. The complete system consists of physical mannequins and models of imaging equipment that link to a computing system and PACS. The computing system carries out simulations in real time according to the setting of the mannequin and the imaging equipment model while the PACS displays and archives the synthetic images.

Monte Carlo simulations are inherently time-consuming but parallel processing is available even in a desktop computer nowadays. With parallelized Monte Carlo codes to take advantage of the multicore processors, the time required to generate a synthetic X-ray image can be achieved within seconds. Real-time online simulation in the system is a reality. The computational phantom is intrinsically linked to the physical mannequin in shape and dimensions. It is possible to create mannequins and computational phantoms representing patients of different size, shape, gender, and age. The presence of the physical model enhances the realism. The students can see and feel the "patient" and the imager and correlate patient positioning with the final image.

Together with dose calculations and tutor's immediate feedback, the proposed system will give students interactive and realistic learning experience. Most importantly, students can experiment with a practically unlimited number of possibilities without any risk or fear of radiation exposure. Extension of the system to other radiological science disciplines can also be achieved easily because the computation engines (Monte Carlo and others) remain the same regardless the application. A patent of the system has been granted.[46]

References

[1] Ziv A, Wolpe PR, Small SD, Glick S. "Simulation-Based Medical Education: An Ethical Imperative," *Simulation in Healthcare*. 2006; 1(4): p 252-256.

[2] Ziv A, Small SD, Wolpe PR. Patient safety and simulation-based medical education. *Medical Teacher*. 2000; 22(5): p 489-95.

[3] Cumin D, Merry AF, Weller JM. Standards for simulation. *Anaesthesia.* 2008; 63(12): p 1281-4.

[4] Issenberg SB, et al.Simulation technology for health care professional skills training and assessment. *Journal of the American Medical Association.* 1999; 282(9): pp. 861-6.

[5] Kneebone RL, Scott W, Darzi A, Horrocks M, Simulation and clinical practice: strengthening the relationship. *Medical education*. 2004; 38(10): p 1095-102.

[6] Gorman PJ,Meier AH, Rawn C, Krummel TM. The future of medical education is no longer blood and guts, it is bits and bytes. *American Journal of Surgery*. 2000; 180(5): p 353-6.

[7] Dawson SL, Kaufman JA. The imperative for medical simulation. *Proceedings of the IEEE*. 1998; 86(3): p 479-483.

[8] Lane JL, Slavin S, Ziv A. Simulation in Medical Education: A Review. *Simulation & Gaming*. 2001; 32(3): p 297-314.

[9] Vozenilek J, Huff JS, Reznek M, Gordon JA. See one, do one, teach one: advanced technology in medical education. *Academic Emergency Medicine*. 2004; 11(11): p 1149-54.

[10] Early S, Roche-Nagle G. Virtual reality technology and surgical training--a survey of general surgeons in Ireland. *Irish journal of medical science*. 175(1): p 15-9.

[11] "Kyotokagaku Co., Ltd." [Online]. Available: http://www.kyotokagaku.com/index.html. [Accessed: 02-May-2011].

[12] Thakur Y, Nikolov HN, Gulka IB, Holdsworth DW, Drangova M. Design and construction of a multipath vessel phantom for interventional training. *British Journal of Radiology*. 2010; 83(995): p 979-982.

[13] "Schallware Ltd." [Online]. Available: http://www.schallware.de/english/index.html. [Accessed: 02-May-2011].

[14] "MedSim | Advanced Ultrasound Simulation." [Online]. Available: http://www.medsim.com/. [Accessed: 16-Mar-2012].

[15] Issenberg SB, McGaghie WC, Petrusa ER, Gordon DL, Scalese RJ, Features and uses of high-fidelity medical simulations that lead to effective learning: a BEME systematic review. *Medical Teacher*. 2005; 27(1): p 10-28.

[16] Young L, Wilkinson D, Futuristic medical education. *Medical Journals of Australia*. 2005; 183(11/12): p 590-591.

[17] Schiavenato M. Reevaluating Simulation in Nursing Education: Beyond the Human Patient Simulator. *Journal of Nursing Education*. 2009; 48(7): p 388-394.

[18] Desser TS. Simulation-based training: the next revolution in radiology education?. *Journal of the American College of Radiology : JACR*. 2007; 4(11): p 816-24.

[19] Gould D. Using simulation for interventional radiology training. *British Journal of Radiology*. 2010; 83(991): p 546-553.

[20] Towbin AJ, Paterson BE, Chang PJ, Computer-based Simulator for Radiology: An Educational Tool. *Radiographics*. 2008; 28(1): p 309 -316.

[21] "Shaderware." [Online]. Available: http://www.shaderware.com/. [Accessed: 02-May-2011].

[22] Levine R. Medical Instruction Using a Virtual Patient. U.S. Patent WO Patent WO/2004/029911Apr-2004.

[23] Wilkins JD. Orthopedic Procedures Training Simulator. U.S. Patent WO Patent WO/2006/1106292006.

[24] Motion Capture Society. [Online]. Available: http://www.motioncapturesociety.com/. [Accessed: 02-May-2011].

[25] Motion Capture Resources. [Online]. Available: http://www.motion-capture-system.com/resources/. [Accessed: 02-May-2011].

[26] Gardner R, Raemer DB. Simulation in obstetrics and gynecology. *Obstetrics and Gynecology Clinics of North America*. 2008; 35(1): p 97-127.

[27] Gordon MS. Cardiology patient simulator: development of an automated manikin to teach cardiovascular disease. *American Journal of Cardiology*. 1974; vol 34: p 350-5.

[28] "Gas Man." [Online]. Available: http://www.gasmanweb.com/. [Accessed: 16-Mar-2012].

[29] Gaba DM,DeAnda A. Acomprehensive anesthesia simulator environment: re-creating the operating room for research and training. *Anesthesiology*. 1988; vol 69: p 387–94.

[30] Cooper JB, Taqueti VR. A brief history of the development of mannequin simulators for clinical education and training. *Postgraduate Medical Journal*. 2008; 84(997): p 563-70.

[31] Guizzo E. Anatomy of a Crash-Test Dummy. *IEEE Apectrum*. 2007.

[32] Caon M, Bibbo G, Pattison J. An EGS4-ready tomographic computational model of a 14-year-old female torso for calculating organ doses from CT examinations. *Physics in medicine and biology*. 1999; 44(9): p2213-25.

[33] Bozkurt A, Chao TC, Xu XG. Organ dose conversion coefficients for 0.1-10 MeV electrons calculated for the VIP-Man tomographic model. *Health physics*. 2001; 81(2): p 203-214.

[34] Zankl M. The GSF Voxel Computational Phantom Family. In *Handbook of Anatomical Models for Radiation Dosimetry*. Xu XG, Eckerman KF, Eds. Boca Raton, USA: CRC Press, 2010, p 65-85.

[35] Zubal IG, Harrell CR, Smith EO, Rattner Z, Gindi G, Hoffer PB. Computerized three-dimensional segmented human anatomy. *Medical physics*. 1994; 21(2): p 299-302.

[36] Alghamdi AA, Ma A, Tzortzis M, Spyrou NM. Neutron-fluence-to-dose conversion coefficients in an anthropomorphic phantom. *Radiation Protection Dosimetry*. 2005; 115(1-4): p 606-11.

[37] Nagaoka T, Watanabe S. Voxel-based variable posture models of human anatomy. *Proceedings of the IEEE*. 2009; 97(12): p 2015-2025.

[38] Fanti V, Marzeddu R, Massazza G, Randaccio P, Brunetti A, Golosio B. A Simulator for X-ray images. *Radiation Protection Dosimetry*. 2005; 114(1-3): pp. 350-4.

[39] James DL, Twigg CD. Skinning mesh animations. *ACM Transactions on Graphics*. 2005; 24(3): p 399-407.

[40] Lee C, Lodwick D, Hasenauer D, Williams JL, Lee C, Bolch WE. Hybrid computational phantoms of the male and female newborn patient: NURBS-based whole-body models. *Physics in medicine and biology*. 2007; 52(12): p 3309-33.

[41] Xu XG, Taranenko V, Zhang J, Shi C. A boundary-representation method for designing whole-body radiation dosimetry models: pregnant females at the ends of three gestational periods—RPI-P3, -P6 and -P9. *Physics in Medicine and Biology*. 2007; 52(23): p 7023-7044.

[42] DeMaria S, Levine AI, Bryson EO. The use of multi-modality simulation in the retraining of the physician for medical licensure. *Journal of Clinical Anesthesia*. 22(4): p 294-9.

[43] Ma AK, Gunn S, Darambara DG. Introducing DeBRa: a detailed breast model for radiological studies. *Physics in Medicine and Biology*. 2009; 54(14): p 4533-45.

[44] Bliznakova K, Bliznakov Z, Bravou V, Kolitsi Z, Pallikarakis N. A three-dimensional breast software phantom for mammography simulation. *Physics in Medicine and Biology*. 2003; 48(22): p 3699-719.

[45] Näppi J, Dean PB, Nevalainen O, Toikkanen S. Algorithmic 3D simulation of breast calcifications for digital mammography. *Computer Methods and Programs in Biomedicine*. 2001; 66(1): p 115-24.

[46] Masar Scientific Ltd Company. Patent number GB2484355 (A) -" *System and method for Radiological simulation"*. UK intellectual property office publication, 11 April 2012.

A Soft Computing Model for Server Outage Detection

Matthias Wastian[1*,2], Michael Landsiedl[1], Felix Breitenecker[2]

[1]Technical Solutions, dwh GmbH, Neustiftgasse 57-59, 1070 Vienna, Austria *matthias.wastian@dwh.at
[2]Inst. of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Haupstraße 8-10, 1040 Vienna

Simulation Notes Europe SNE 25(1), 2015, 27 - 34
DOI: 10.11128/sne.25.tn.10277
Received: September 10, 2014; Revised January 15, 2015;
Accepted: March 10, 2015;

Abstract. Several approaches to detect or even predict abnormal events as early as possible will be discussed. The model input is a time series of frequently collected data. The approaches presented in this document use various methods originating in the field of data mining, machine learning and soft computing in a hybrid manner. After a basic introduction including several areas of application, the focus will lie on the modular parts of the proposed server outage model, starting with a discussion about different approaches to time series prediction such as SARIMA models and specific artificial neural networks. After the presentation of several algorithms for outlier detection (angle-based outlier factor, one-class support vector machines) the gained results of the simulation are put up for discussion. The text ends with an outlook for possible future work.

Introduction

Before we want to discuss abnormal event detection in general, we state the following two definitions.

Definition 1 (Event): *An event shall be defined as an occurrence happening at a determinable time and place with a certain duration. It may be a part of a chain of occurrences as an effect of a preceding occurrence and as the cause of a succeeding occurrence. It is possible that more than one event occurs at the same time and/or place.*

Definition 2 (Abnormal Event): *An abnormal event shall be defined as an outlier in a chain of events, an event that deviates so much from the other events as to arouse suspicion that it was caused by something that does not follow the usual behavior of the considered system and that it could change the entire system behavior.*

Applications of abnormal event detection can be found in a broad variety of areas, almost all of them following the idea to guarantee a certain level of safety for the system considered. Examples are the prediction or detection of server outages, of natural catastrophes like flooding, hurricanes or earthquakes, of stock market breakdowns and of network intrusions. In the area of audio and video surveillance crowd behavior or traffic might be analyzed, but abnormal event detection also plays an important role in ambient assisted living.

Various approaches have been suggested for abnormal event detection. This paper is going to focus on time series forecasting with artificial neural networks (ANN) and outlier detection of the prediction errors with one-class support vector machines (OC-SVM) as proposed by [4], [5], [6], [7] as well as by [8]. Other applied methods in the field of abnormal event detection are listed below:

- sparse reconstruction cost ([14])
- wavelet decomposition ([15])
- clustering based abnormal event detection ([12])
- change point detection ([11])
- explicit descriptors statistical model
- bayes estimation
- maximum likelihood
- correlation analysis
- principal component analysis (PCA).

1 Data Generation and Data Preprocessing

1.1 Data generation

Server monitoring is rampant nowadays. Server monitoring software allows to measure lots of features of a server that somehow describe its status. For our simulations, we had a total of up to 1439 features per server which were measured at a sampling rate from about one per fifteen minutes up to one per minute.

Besides historic data sets of several servers that were logged in the past, IBM Lotus Domino Server.Load was used to generate artificial data sets. The capacity-planning tool was used to run tests, also called scripts and workloads, against a targeted server to measure its server capacity and response metrics. During these tests, each client generated a simulated user load of transactions against the server under test, which reported server statistics back to the client.

1.2 Data preprocessing

First of all, the size of the recorded data set is rather large. All the simulations for a rapid server alert system have to be carried out at least nearly online. Thus a reduction of the original data set is indispensable. We used expert knowledge and did a feature selection by categorizing the features into four groups of different priorities, resulting in up to 14 features of the highest priority 0 and up to 73 features of the two most important priorities 0 and 1. Most simulation runs were implemented using the data labelled with these two priorities.

As the model intends to recognize the actual and future status of a server, those features that accumulate values (e.g., number of mails sent since the start of the server monitoring) were transformed into their differences.

Wrong measurements are also an issue that has to be dealt with for the server outage detection model. Especially features that deal with the queue length of hard disks delivered impossible values in a few cases. These values were substituted by their predecessors (if those were possible values) during the learning process. Of course, this substitution is also possible during on-line simulation runs.

Another possibility is to delete those wrong values like it needs to be done, when a measurement cannot be carried out correctly due to any reason and the feature at this time is NaN. The distribution of these NaNs can be investigated separately. The algorithms proposed in the following sections are not able to deal with NaNs.

The ranges of the features considered in the model differ a lot. To make them comparable, the whole data set needs to be normalized. When using the neuro-predictor for the rapid server alert model, is seems best to use the following minmax-mapping to normalize the data:

$$f(x) = y_{min} + \frac{(y_{max} - y_{min})(x - x_{min})}{(x_{max} - x_{min})} \qquad (1)$$

This is an affine transformation from $[x_{min}, x_{max}]$ to $[y_{min}, y_{max}]$.

2 Predictor

Given any process that is checked for abnormal events, usually some features of this process can be measured at a constant sampling rate. Let m be the number of observed features. This results in m univariate time series. Given some past values and the actual value x_n of a certain feature, it is possible to predict the next observation x_{n+1} with a predictor and to calculate the prediction error as soon as the true new value x_{n+1} is measured.

Besides the classic ARIMA models that can be used for time series prediction, a certain kind of ANNs has proven to be an efficient predictor. Both models are going to be introduced in the following subsections. A multivariate approach is not recommended based on the simulation results for the server outage prediction as well as based on the results of various other authors. If a multivariate approach is desired nevertheless, we suggest to cluster the features first into several groups and to use an own multivariate predictor for each group.

The basic idea for any predictor of the abnormal event detection model is that the predictions are very good, if there are no abnormal events, i.e., the system's status is normal. The predictions become worse and do not originate from the usual distribution at least at the beginning of an abnormal event.

From a time series point of view, the most difficult task for the predictor is to consider the seasonality of the time series of some features.

For example, the number of logged in users of a company on a certain Monday at 9:00 a.m. will probably strongly depend on the number of logged in users on Monday one week before at the same time. Feasts and holidays can cause problems for such models.

2.1 Neuro-Predictor

ANNs are non-linear and data-driven by nature and therefore at least theoretically very well suited to model seasonality interacting with other components.

[16] refers to Simon Haykin, who suggests choosing the number of training patterns based on

$$N = \frac{W}{\varepsilon} \tag{2}$$

W shall be the number of weights used in the ANN, ε shall be the error the training examples should be classified with and N shall be the number of patterns in the training set in this context.

When using ANNs to forecast time series, data normalization is a key issue. Various normalization methods can be applied; logarithmic or exponential scaling can be used if problems with non-linearities are expected during the network training. Linear normalizations like (1) can be used to meet the requirements of the network input layer, as the input range must not be too wide.

Significant patterns as seasonality and trends should be removed, if possible, to make the ANN time series model easier. To be able to use the concept of cross-validation, appropriate training, test and validation data sets need to be chosen. For our simulations the training data includes 70%, the test and the validation set includes 15% of the preprocessed data each.

The tasks of structuring the data and choosing the number of input nodes n_i of the ANN predominantly depend on the number d of lagged values to be used for forecasting of the next value in the standard case of a one-step-ahead prediction. Thus the function to be modeled by the ANN is of the type

$$x_{n+1} = f(x_n, x_{n-1}, \dots, x_{n-d+1}) \tag{3}$$

This function can also be alternated to

$$x_{n+1} = f(x_n, x_{n-1}, \dots, x_{n-d+1}, x_{n-s}, \dots, x_{n-2s}, \dots) \tag{4}$$

for a seasonality s. If the seasonality was not removed and the data preprocessing produces suitable input data blocks, seasonality can thus be modeled in an explicit way by the neuro-predictor.

The number of output neurons n_o directly corresponds to the forecasting horizon, i.e. in the case of a one-step-ahead forecast there is only one output neuron. Usually only one hidden layer is used. The number of the neurons in the hidden layer n_h was chosen according to the geometric pyramid rule:

$$n_h = \alpha\sqrt{n_i n_o}, \;\; \alpha \in [0.5,2] \tag{5}$$

Choosing the number of hidden neurons as well as the data normalization involves trial-and-error experimentation.

We used the hyperbolic tangent as activation function in the hidden layer (the sigmoid function is also possible) and the linear activation function for the output layer. According to [2], a non-linear activation function in the output layer is only needed, if the time series shows a significant trend even after the data preprocessing.

For the training of such neuro-predictors we use the Levenberg-Marquardt algorithm. The training sets are presented to the ANNs in several epochs. The supervised learning stops as soon as one of the following three break conditions is met:

1. The number of training epochs exceeds the value of a chosen tuning parameter.
2. The number of back-to-back epochs, which the error function of the validation set increases in, exceeds the value of a chosen tuning parameter.
3. The error value of the test data set falls below some minimal error value (e.g. 10^{-6}).

If there are several ANN models that we can finally choose from, an adapted version of the AIC can be applied:

$$AIC = Nn_o \ln(\sigma^2) + 2k \tag{6}$$

The model with the smallest AIC shall be preferred.

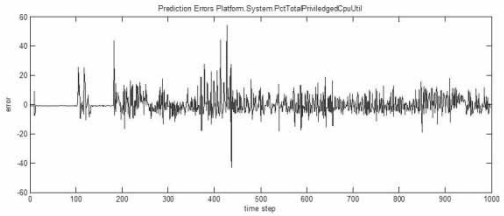


Figure 1. Prediction errors of a certain server feature, using a neuro-predictor.

2.2 SARIMA Models

B being the backshift operator, autoregressive integrated moving average models with parameters p, d and q for a time series $\{x_t\}$ with error terms $\{\varepsilon_t\}$ are given by

$$\phi(B)x_t = \theta(B)\varepsilon_t \tag{7}$$

with

$$\phi(B) = \left(1 - \sum_{i=1}^{p} \phi_i B^i\right)(1-B)^d \tag{8}$$

and

$$\theta(B) = 1 - \sum_{i=1}^{q} \theta_i B^i. \tag{9}$$

If the time series exhibits a strong seasonality, the model is adapted to a seasonal autoregressive integrated moving average model with parameters $(p,d,q) \times (P,D,Q)_s$, which is given by

$$\Phi(B^s)\varphi(B)\nabla_s^D \nabla^d x_t = \Theta(B^s)\theta(B)\varepsilon_t \tag{10}$$

with ∇ being the differencing operator, D the number of seasonal differences, Φ a polynomial of degree P, Θ a polynomial of degree Q and

$$\varphi(B) = \left(1 - \sum\nolimits_{i=1}^{p} \phi_i B^i\right). \tag{11}$$

First of all, the orders of differencing have to be identified to attain a stationary time series, several transformations like the logarithmic one might be useful. By looking at the plots of the autocorrelation function (ACF) and the partial autocorrelation function (PACF) – they are in fact bar charts – of the differenced series, the numbers of AR and/or MA terms that are needed can tentatively be identified, for example following the advices that can be found in [1].

2.3 Comparison Between Neuro-Predictors and SARIMA Models

When using ANNs for prediction, the results obtained by various authors differ widely in quality: Some suggest that ANNs are better than other forecasting models, others contradict them. Some have seemed to obtain better results with seasonally adjusted data, others think that ANNs are able to directly model seasonality in an implicit way, without any seasonal adjustments on the input data. Detailed research results are presented in [2].

In 1991, Sharda, Patil and Tang identified a number of facts that determine which method is superior by experiments:

- For time series with long memory, both approaches deliver similar results.
- For time series with short memory, ANNs outperform the traditional Box-Jenkins approach in some experiments by more than 100%.
- For time series of various complexities, the optimally tuned neural network topologies are of higher efficiency than the corresponding traditional algorithms. [16]

A hybrid combination of neural networks and traditional approaches – maybe also including GARCH models – seems very promising.

For the server outage detection model, some time series involved might have a long memory, others a short one. All in all, it seems reasonable that it is less inexact to choose the same parameters for all the feature predictors, if the neuro-predictors are used. Choosing the same parameters for all the predictors simplifies the model a lot.

General Model Assumption.

The predictors work in a rather exact way, if and only if the server status is ok.

3 Outage Detector

An analysis of prediction errors is the basis for the anomaly detector. The outage detector decides in a multivariate way, whether the prediction errors of all the features belong to the class ‚normal' or not. We did not only let the anomaly detector decide upon the most recent prediction errors, but we also made him judge upon a moving average of the prediction errors, which increases the tolerance against weaknesses within the prediction models.

Depending on the number of features predicted, the dimension of the prediction error vector is a key issue for choosing a good anomaly detector. For increasing dimension the relevance of distance converges against 0 – a phenomenon which is part of the curse of dimensionality.

[17] distincts three fundamental approaches to detect outliers:

- Model neither normality nor abnormality. Determine the outliers with no prior knowledge of the data. This is essentially a learning approach analogous to unsupervised clustering.
- Model both normality and abnormality. This approach is analogous to supervised classification and requires pre-labeled data, tagged as normal or abnormal.
- Model only normality; maybe tolerate abnormality in very few cases. Authors generally name this technique novelty detection or novelty recognition, especially if only normal data is given. It is analogous to a semi-supervised recognition or detection task. Only the normal class is taught but the algorithm learns to recognize abnormality. Theapproach needs preclassified data but only learns data marked normal.

3.1 Threshold

For lower dimensions a simple threshold for a prediction error norm like the Euclidean norm can be sufficient to detect anomalies (assuming that all the features have been transformed to similar ranges during the preprocessing). If the predictions of several features are as bad as the ones on the outside margin of the Gaussian bell of figure 2, they will be detected by simple threshold.

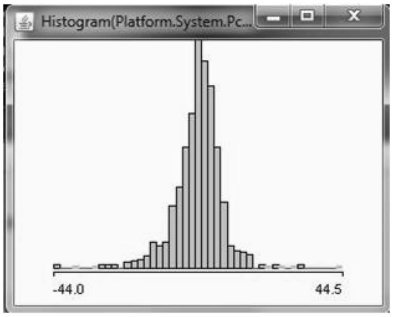


Figure 2. A typical histogram of the prediction errors of a single server feature: A Gaussian bell and a few outliers clearly visible on the outside margin

3.2 Angle-Based Outlier Detection

Angles are more stable than distances in high-dimensional spaces, which suggests the use of angles instead of distances for high-dimensional data. In fact, the situation is contrary for low-dimensional data. The angle-based outlier detection (ABOD) method alleviates the effects of the notorious curse of dimensionality compared to purely distance-based methods.

Following the idea of the algorithm developed by Kriegel, Schubert and Zimek (2008, see [9]), a point is considered as an outlier, if most other points are located in a similar direction, and a point is considered as an inlier, if many other points are located in varying directions. The broadness of the spectrum of the angles between a certain point A and all pairs of the other points is a score for the outlierness of A: The smaller the score, the greater is the point's outlierness. The idea of the algorithm is illustrated for two dimensions in figure 3.

The angles in the so-called angle-based outlier factor are weighted by the squared inverse of the corresponding distances to avoid bigger problems with low-dimensional data sets.

$$ABOF(A) = VAR_{B,C\,\epsilon\,D}\left(\frac{\langle AB, AC\rangle}{\|AB\|^2\|AC\|^2}\right) \qquad (12)$$

A possibility to approximate the computationally expensive ABOF is to calculate the variance of the angles only of the pairs of points which belong to the k nearest neighbors of A, since these are the ones with the largest weights in the formula (12). [10] provides further details on this issue.

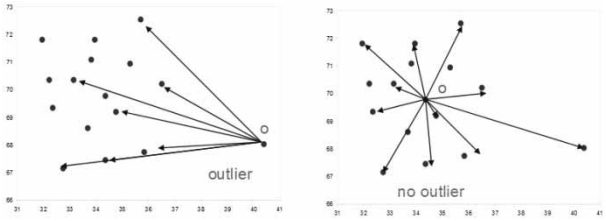


Figure 3. Idea of angle-based outlier detection

3.3 One-Class Support Vector Machine

In general, one-class support vector machines (OC-SVMs) are designed for the certain type of a $(1 + x)$-class learning task. This is a model with an unknown number of classes, but the modeler is only interested in one specific class. Typical examples for these kinds of tasks are content-based image retrieval or document-retrieval in general. Making research for this paper on the internet can be seen as such a task: Papers which treat relevant topics are alike, they represent the class the modeler is interested in. These are the positive examples and it is easy to find some good representatives of this class. The negative examples are simply the rest of the web pages or papers, and they originate from an unknown number of different negative classes.

It is daunting and wrong to try to characterize the distribution of the negatives in such cases; they could belong to any negative class, and the modeler is not

even interested which exact negative classes they might belong to. Each negative example is negative in its own way, but as the positive ones are alike, it is possible to model their distribution. According to this the OC-SVM is a typical example of a model of normality, matching the third approach described at the beginning of this section.

The OC-SVM tries to fit a tight hypersphere W to include most, but not all positive examples. If it is attempted to fit all positive examples, this would lead to overfitting. In fact, the OC-SVM searches for the maximal margin hyperplane

$$\omega x + b = 0 \tag{13}$$

with a normal vector ω and a bias b which separates the training data from the origin in the best way. It may be interpreted as a regular two-class SVM, where almost all the training data lies in the first class and the origin is the only member of the second class.

If the one class the modeler is interested in is considered as the regular data, resulting from normality, the negative examples detected by the OC-SVM can be considered as outliers of a different nature resulting from anomaly. This makes the OC-SVM an effective outlier detection tool.

Let $\{x_1, \ldots, x_n\}, x_i \in X \subseteq \mathrm{R}^m$ be a training set of $n \in \mathrm{N}$ observations that belong to a single class. The OC-SVM aims to define the minimum volume region enclosing $(1-\nu)n$ observations. The parameter $\nu \in [0,1]$ thus controls the fraction of observations that are allowed to be outliers. K shall be a kernel with a mapping function φ. ξ_i shall be the slack variables for observations on the wrong side; non-zero slack variables correspond to the tolerated outliers. The OC-SVM algorithm results in the following minimization problem:

$$\min_{\omega,\xi,b} \frac{1}{2}\|\omega\|^2 - b + \frac{1}{\nu n}\sum_{i=1}^{n} \xi_i \tag{14}$$

subject to

$$\omega^T \varphi(x_i) - b \geq \xi_i \geq 0 \tag{15}$$

Solving the OC-SVM optimization problem is equivalent to a dual quadratic programming problem with Lagrangian multipliers α_i that can be solved with standard methods:

$$\max_{\alpha_i} -\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j K(x_i, x_j) \tag{16}$$

subject to

$$\sum_{i=1}^{n} \alpha_i = 1, \;\; 0 \leq \alpha_i \leq \frac{1}{\nu n} \tag{17}$$

Those patterns with corresponding $\alpha_i > 0$ are the support vectors. By using the Karush-Kuhn-Tucker conditions ω and b can be obtained as

$$\omega = \sum_{i=1}^{n} \alpha_i x_i \tag{18}$$

$$b = \sum_{i=1}^{n} \alpha_i x_i^T x_j \tag{19}$$

for any support vector x_j.

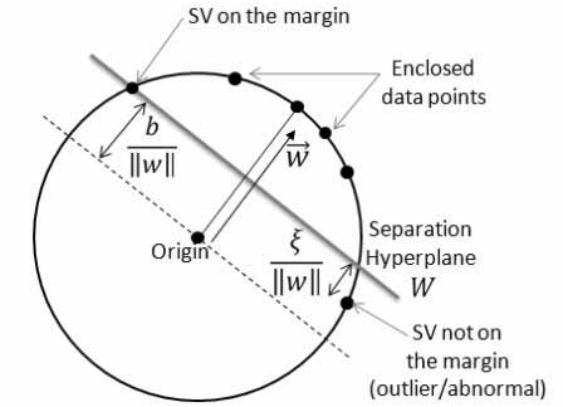


Figure 4. One-class support vector machine [8]

A new observation x is labeled by the OC-SVM via the decision function

$$f(x) = \omega^T \varphi(x) - b \tag{20}$$

which is positive for inliers and negative for outliers.

According to [8], it is easily possible to define a family of decision rules introducing a threshold $\gamma \in \mathrm{R}$ by using an adaption of (20) and dividing inliers and outliers along γ instead of 0. This formulation allows controlling the trade-off between the probability to miss outliers and the probability to falsely declare an observation an outlier.

3.4 Combined Detector

As all the proposed outlier detector methods return an outlierness score for a feature vector, they could be used in a hybrid way. Then a weighted sum of the outlierness scores of each method is the final outlierness score of an observation. The ideas to compare outlier scores provided by [19] should be obeyed.

4 Results and Outlook

First of all, it has to be stated that it is almost impossible to precisely define the term server outage, wherefore a definition is not given in this paper. Any limitation to the normal operation of a server is unwanted. Many times only a certain kind of tasks is delayed or cannot be executed at all. The severity of this limitation also depends on the fact whether users can carry out other tasks in the mean time. The only possibilities to give the modeler an idea about the severity of an outage are the total downtime minutes or downtime minutes per user. Thus the basic idea of this model is to be able to provide the administrator of a server with the detection/prediction of irregularities, of anomalies which differ from the usual server operation. A classification of outages would be very useful, but requires labelled outage data to learn from. This remains future work.

Within the proposed model, the numbers of lagged time series elements that are relevant for the univariate prediction models for each server feature are not very easy to determine and the optimal number probably varies for each variable. Also the seasonality of the feature time series is not easy to diagnose. Nevertheless, the prediction models with global parameters for all the predictors worked very well during a normal operation of servers and seem to be sufficient for an online server outage detection model.

During several test runs, the anomaly detectors easily detected when the servers changed their status from idle to busy and vice versa (see figure 5). They also detected abnormal events within the gas price time series which was used as a benchmark data set (see figure 6). For this time series, an abnormal event is for example the oil crisis of 1979, which was caused by the Islamic revolution in Iran and the first gulf war, i.e. by external events. For the server outage detection model, the verification is rather difficult and there will be done further research on this topic: Besides the difficulty to define a server outage, the model needs to be tested in a real-life scenario which is planned in near future. So far, the detectors worked well with the test data sets.

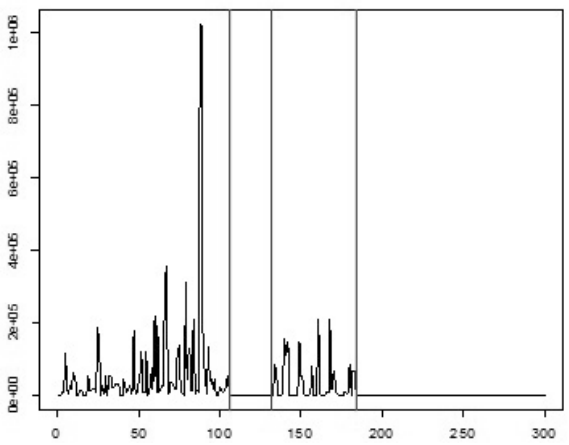


Figure 5. Angle-based outlier detector detecting the server change from idle to busy (green) and busy to idle (red)

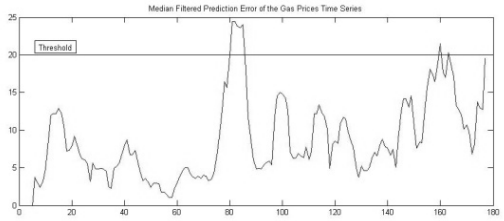


Figure 6. Median-filtered prediction error of the gas prices time series using a neuro-predictor with a delay of 3 months, 10 hidden neurons and a threshold for abnormal event detection. The median was calculated over 6 months. The first peak above the threshold 20 corresponds to the 1979 oil crisis.

Of course, a server outage prediction software has a cold start: During the training some internal model parameters that are required to run the model need to be adjusted, before an expert can adjust several tuning parameters to control the alert sensitivity of the software. The most important tuning parameters are part of the anomaly detector. One could say that the server outage detection model needs to get to know the server that the outages shall be predicted of. As parts of the model are able to learn from the past, the software is expected to highly improve its performance after several days.

An important question that still remains unanswered is when the neuro-predictors should be retrained or when the ARIMA models should be updated. Certainly, if the way the server is used changes considerably, a restart of the model is necessary.

References

[1] Nau R. Forecasting – Decision 411, online, available at people.duke.edu/~rnau/411home.htm, 2005.

[2] Zhang GP, Kline D. *Quarterly Time-Series Forecasting with Neural Networks*. In: IEEE Transactions on Neural Networks, IEEE Computational Intelligence Society. 2007; 8(6): p 1800 - 1814.

[3] Crone S, Dhawan R. *Forecasting Seasonal Time Series with Neural Networks: A Sensitivity Analysis of Architecture Parameters*. In: Proc. International Joint Conference on Neural Networks 2007, IEEE, Orlando, Florida; 2007. p 2099 - 2104.

[4] Heller K, Svore K, Keromytis A, Stolfo S. *One Class Support Vector Machines for Detecting Anomalous Windows Registry Accesses*. In: Proc. Workshop on Data Mining for Computer Security, IEEE International Conference on Data Mining 2003, Melbourne, Florida; 2003. p 2 - 9.

[5] Evangelista P, Bonnisone P, Embrechts M, Szymanski B. *Fuzzy ROC Curves for the 1-Class SVM: Application to Intrusion Detection*. In: Proc. 13th European Symposium on Artificial Neural Networks 2005, d-side, Bruges, Belgium; 2005. p 345 - 350.

[6] Zhang R, Zhang S, Lan Y, Jiang J. *Network Anomaly Detection Using One Class Support Vector Machine*. In: Proc. MultiConference of Engineers and Computer Scientists 2008, Volume 1, IAENG, Hong Kong, 2008.

[7] Dreiseitl S, Osl M, Scheibböck C, Binder M. *Outlier Detection with One-Class SVMs: An Application to Melanoma Prognosis*. In: Proc. AMIA Annual Symposium 2010, 2010; p 172 - 176.

[8] Lecomte S, Lengellé R, Richard C, Capman F, Ravera B. *Abnormal Events Detection Using Unsupervised One-Class SVM – Application to Audio Surveillance and Evaluation*. In: Proc. 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance 2011, IEEE, Klagenfurt, Austria, 2011; p 124 - 129.

[9] Kriegel HP, Schubert M, Zimek A. *Angle-Based Outlier Detection in High-Dimensional Data*. In: Proc. 14th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining 2008, Las Vegas, Nevada, ACM, New York, 2008; p 444 - 452.

[10] Pham N, Pagh R. *A Near-Linear Time Approximation Algorithm for Angle-Based Outlier Detection in High-Dimensional Data*. In: Proc. 18th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining 2012, ACM, New York, USA, 2012; p 877 - 885.

[11] Guralnik V, Srivastava J. *Event Detection from Time Series Data*. In: Proc. 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, USA; 1999. p 33-42.

[12] Jiang F, Wu Y, Katsaggelos A. *Abnormal Event Detection Based on Trajectory Clustering by 2-Depth Greedy Search*. In: Proc. IEEE International Conference on Acoustics, Speech and Signal Processing 2008, IEEE, Las Vegas, Nevada, USA; 2008. p 2129 - 2132.

[13] Hawkins S, He H, Williams G, Baxter R. *Outlier Detection Using Replicator Neural Networks*. In: Proc. 4th International Conference on Data Warehousing and Knowledge Discovery 2002, Aix-en-Provence, France, Lecture Notes in Computer Science 2454, Springer; 2002. p 113 - 123.

[14] Cong Y, Yuan J, Liu J. *Sparse Reconstruction Cost for Abnormal Event Detection*. In: Proc. 24th IEEE Conference on Computer Vision and Pattern Recognition, IEEE, Colorado Springs, Colorado, USA; 2011: p 3449 - 3456.

[15] Suzuki M, Ihara H. *Development of Safeguards System Simulator Composed of Multi-Functional Cores*. Journal of Power and Energy Systems. Volume 2, Number 2, J-Stage, Japan; 2008. p 899 - 907.

[16] Palit A, Popovic D. Computational Intelligence in Time Series Forecasting – Theory and Engineering Applications. Springer, London; 2005.

[17] Hodge V, Austin J. *A Survey of Outlier Detection Methodologies*. Artificial Intelligence Review. Kluwer Academic Publishers, Netherlands, 2004; 22(2). p 85-126.

[18] Schölkopf B, Smola A. Learning with Kernels. MIT Press, Cambridge, Massachusetts, USA; 2002.

[19] Kriegel HP, Kröger P, Schubert E, Zimek A. *Interpreting and Unifying Outlier Scores*. In: Proc. 11th SIAM International Conference on Data Mining, Mesa, Arizona, USA; 2011.

[20] Aggarwal C, Yu P. *An Effective and Efficient Algorithm for High-Dimensional Outlier Detection*. The VLDB Journal 14, Springer; 2005: p 211-221.

A Comparison of Microscopic Pedestrian Simulation Models based on RiMEA Test Cases

Stefan Seer*, Thomas. Matyus

Mobility Department, AIT Austrian Institute of Technology, Giefinggasse 2, 1210 Vienna, Austria
*stefan.seer@ait.ac.at

Simulation Notes Europe SNE 25(1), 2015, 35 - 42
DOI: 10.11128/sne.25.tn.10279
Received: January 10, 2015; Revised March 10, 2015;
Accepted: March 31, 2015;

Abstract. Simulations enable to predict pedestrian flows for the evaluation of architectural designs and operational plans. In order to assess the strength and weaknesses of different pedestrian simulation models, their performance has to be evaluated in a qualitative and quantitative manner. The RiMEA-Guideline aspires to define a minimum standard for evaluation analysis based on different test cases for evaluating implementations of pedestrian simulation models. This paper provides a comparison of three different pedestrian simulation models, i.e. Social Force, Cellular Automaton and Optimal Reciprocal Collision Avoidance, based on selected test cases from the RiMEA-Guideline. The results provide model developers and practitioners valuable insights into the major differences between the evaluated pedestrian simulation models.

Introduction

Over the last years, microscopic pedestrian simulation models have proven to be a valuable tool for the prediction of pedestrian flows to evaluate architectural designs and operational plans. These models can simulate detailed behaviour of individual humans and represent collective phenomena such as emergent behaviour.

In order to objectively compare different implementations of microscopic pedestrian simulation models, their performance has to be assessed qualitatively with respect to emerging spatial-temporal patterns (e.g. lane formation) and quantitatively based on evaluation measures with respect to accuracy (e.g. reproducibility of pedestrian densities). As of now, several evaluation measures have been described and used in the literature. One recent attempt to define a minimum standard for evacuation analysis is stated by the development of the RiMEA-Guideline [1] which includes fourteen test cases for evaluating implementations of pedestrian simulation models. In addition the United States' National Institute of Standards and Technology (NIST) recommended a set of seventeen verification tests in order to verify building fire evacuation models [2].

The contribution of this paper is to provide a comparison of three different microscopic pedestrian simulation models based on selected test cases from the RiMEA-Guideline. Therefore, we implemented a Social Force model [3], a Cellular Automaton [4] and an Optimal Reciprocal Collision Avoidance model [5]. The results give valuable insights into the major differences between the evaluated pedestrian simulation models which are important for model developers as well as for practitioners.

The remainder of this paper is structured as follows: Section 1 provides an overview of the related work. Section 2 describes the test cases that were used in this work. Section 3 outlines the models for pedestrian simulation. Section 4 presents the evaluation results from applying the modelling approaches to the test cases. Section 5 summarizes the results and discusses the main outcomes. Section 6 concludes the findings and provides recommendations for future work.

1 Related Work

The RiMEA-Guideline (in German: RiMEA-Richtlinie, Richtlinie für Mikroskopische Evakuierungs Analysen – Guideline for Microscopic Evacuation Analyses; RiMEA-Guideline) is a guideline for German-speaking authorities to evaluate the quality of evacuation analyses for complex buildings. Based on the RiMEA-Guideline expert reports are written to ensure that the fundamental questions of an evacuation analysis are answered.

The RiMEA-Guideline has been used in several scientific contributions for the demonstration and evaluation of pedestrian simulation models: In [6] a dynamic distance potential field method for route choice on the operational level of pedestrian dynamics has been described and was applied in a simulation of a RiMEA test case. In [7] a cellular automaton based on a hexagonal grid was calibrated and the simulation results were evaluated according to a test provided by RiMEA. Furthermore, the results of different commercial simulation tools (e.g. Viswalk, PedGo, ASERI) with respect to the RiMEA test cases are published on the RiMEA Website [8].

In [9] the tests recommended by NIST were simulated using the PEDFLOW tool, which lead to the identification of several shortcomings and modifications for further improvements of the tool.

2 Description of Test Cases

The RiMEA-Guideline [8] includes a description of different test cases for evaluating implementations of pedestrian simulation models to reproduce a set of requirements for an evacuation analysis. As of now, 14 test cases are defined in total. In this paper we used the following three test cases for the model comparisons:

- *Test Case 4*: Specific flow through an opening
- *Test Case 6*: Moving around a corner
- *Test Case 12*: Effects based on bottlenecks

The comparison in this work has put the focus on the core functionality of the investigated models only. Hence, we selected test cases which do not include aspects of dynamic routing (e.g. selection of exits). In the following the three test cases used in this study are described in detail.

2.1 Test Case 4

Based on a periodic boundary system with a width of 4 m the specific flow (in persons/ms) should be measured for different densities (in persons/m²) inside the system. The results of this test case should reveal the relation between specific flow and density in a so-called *Fundamental Diagram* [10] as shown in Figure 1.

Since periodic boundaries are hard to implement in a simulator, we use an approximation for this test case by modelling a corridor 60 m in length and 10 m in width as illustrated in Figure 2.

Over the available area, we equally distributed pedestrians and varied their total number in different simulation runs in order to generate average densities of 1, 2, 3, 4 and 5 persons/m². Each pedestrian should move towards the same end of the corridor.

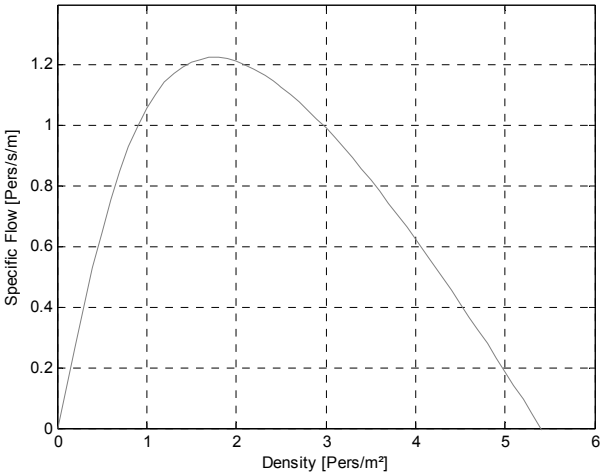


Figure 1. Fundamental diagram representing specific flow
(y-axis) and density (x-axis) based on [10].

The averages of density and velocity are measured in an interval of 1s inside an area of 2 x 2m (see blue rectangle in **Figure 2**) and an area of 4 x 4m (see orange rectangle in **Figure 2**) located at the centre of the corridor. Size and location of the measurement areas have been chosen in order that no boundary effects from walls are measured.

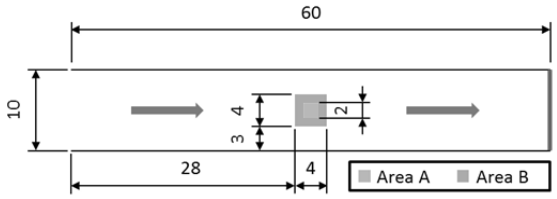


Figure 2. Test Case 4 - Pedestrians are equally distributed over the available area and move towards the right end (red line) of the corridor (red arrows denote walking direction). All measures are in m.

2.2 Test Case 6

In this test case 20 pedestrians should move around a 90° corner without "crossing" walls. The layout of the corner in this test case is illustrated in **Figure 3**. Furthermore, the walking time for each pedestrian is measured between L_{Start} and L_{End}.

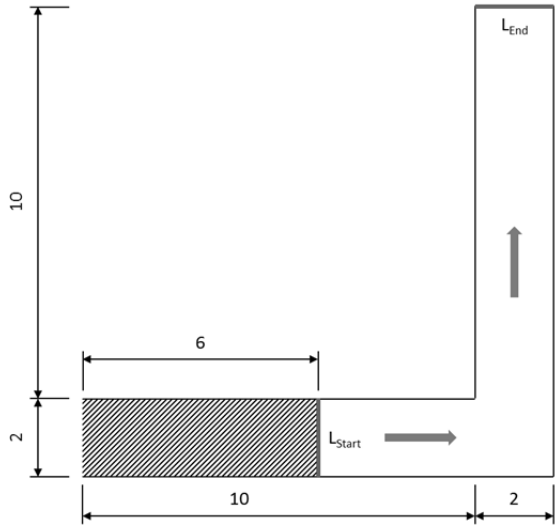


Figure 3. Test Case 6 - Pedestrians are placed in the crosshatched area and walk around the corner (red arrows denote walking direction) without crossing walls and corner. All measures are in m.

2.3 Test Case 12

In this test case 150 pedestrians should be placed in the crosshatched area of the first room (see **Figure 4**) and should move immediately towards the exit in the second room using the connecting corridor. The results should reveal if congestion appears at the exit. Since the pedestrian flow is limited by the bottleneck in the first room, it is expected that congestion only appears at this location and not at the exit in the second room. Therefore, the density is measured for each room in two different areas (see **Figure 4**): Area A (blue) covers a 1x1m region directly in front of each bottleneck and Area B (orange) covers a region of 5m² in the vicinity.

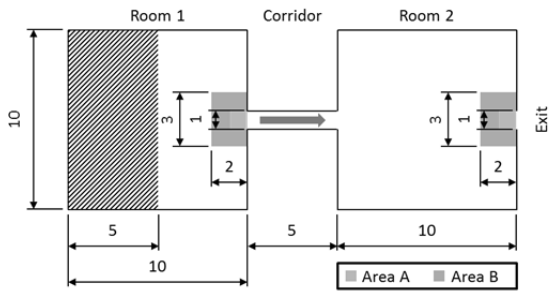


Figure 4. Test Case 12 - The bottleneck in Room 1 should lead to congestion while this should not occur in Room 2. All measures are in m.

3 Modelling Approaches

For the comparisons in this paper, we selected three different microscopic modelling approaches for pedestrian simulations, namely:

- *Social Force* from [3] and [11]
- *Cellular Automaton* from [4]
- *Optimal Reciprocal Collision Avoidance* from [5]

The choice was mainly motivated by the fact that these approaches are widely used within commercial tools and the scientific community.

3.1 Social Force

The Social Force approach introduced by [11] defines attraction and repulsion forces with respect to other humans and the environment, thus representing individual walking behaviour as a sum of different accelerations as

$$f_\alpha(t) = \frac{v_\alpha^0 e_\alpha - v_\alpha}{\tau_\alpha} + \sum_{\beta \neq \alpha} f_{\alpha\beta}(t) + \sum_i f_{\alpha i}(t). \quad (1)$$

The acceleration f_α at time t of an individual α towards a certain goal is defined by the desired direction of movement e_α with a desired speed v_α^0. Here, the current velocity v_α is adapted to the desired speed v_α^0 within a certain relaxation time τ_α. The movement of a pedestrian α is influenced by other pedestrians β which is modelled as a repulsive acceleration $f_{\alpha\beta}$. A similar repulsive behaviour for static obstacles i (e.g. walls) is represented by the acceleration $f_{\alpha i}$. For notational simplicity, we omit the dependence on time t for the rest of the paper.

In this work we use a Java implementation of the definition of an elliptical repulsive force from [3] formulated by

$$f_{\alpha\beta} = a_\alpha e^{-\frac{w_{\alpha\beta}}{b_\alpha}} \frac{d_{\alpha\beta}}{\|d_{\alpha\beta}\|}, \quad (2)$$

where the semi-minor axis $w_{\alpha\beta}$ of the elliptic formulation is defined by

$$w_{\alpha\beta} = \frac{1}{2}\sqrt{\left(\|d_{\alpha\beta}\| + \|d_{\alpha\beta} - (v_\beta - v_\alpha)\Delta t\|\right)^2 - \|(v_\beta - v_\alpha)\Delta t\|^2}. \quad (3)$$

Here, the velocity vectors v_α and v_β of pedestrians α and β are included allowing to take into account the step size of pedestrians. Furthermore, we take into account that pedestrians have a higher response to other pedestrians in front of them by including an anisotropic behaviour, as described in [3].

3.2 Cellular Automaton

In [4] a two-dimensional cellular automaton model is presented for simulating pedestrian movement. Each cell has a size of 40 × 40 cm² and can either be empty or occupied by exactly one pedestrian. The probabilities for moving a pedestrian are encoded in a 3×3 matrix where the central element describes the probability for the pedestrian not to move at all, while the remaining 8 correspond to a move to the neighbouring cells. If a cell is occupied, the probability is set to zero. The update is performed in parallel for all pedestrians and conflicts are resolved according to the following rules: If no other pedestrian targets the desired cell, the move is executed. If more than one pedestrian share the same target cell, one is chosen according to the relative probabilities based on which each pedestrian has chosen the target. The first ranked pedestrian moves while its rivals for the same target keep their position.

Long-range interactions between pedestrians are modelled using a floor field which modifies the transition probabilities to neighbouring cells. This field can be discrete or continuous and is subject to diffusion and decay (e.g. to model the behaviour of following other pedestrians). Furthermore, it is modified by the motion of the pedestrians. Therefore, the model uses an idea similar to chemotaxis, but with pedestrians following a virtual rather than a chemical trace. The results of [4] show that their Cellular Automaton approach is able to model collective and self-organization effects such as lane formation in counterflow through a large corridor.

3.3 Optimal Reciprocal Collision Avoidance

The approach of the Optimal Reciprocal Collision Avoidance (ORCA) model as described in [5] implies that each individual takes into account the observed velocity of other individuals in order to avoid collisions. Individuals are reciprocally collision-avoiding (they "share the responsibility") and it is guaranteed that two particular individuals are collision-free for at least a fixed amount of time into the future.

Thus, for each other individual the model derives a half-plane (in velocity-space) of velocities that are allowed to be selected in order to guarantee collision avoidance. The individual then selects its optimal velocity from the intersection of all permitted half-planes, which can be done efficiently using linear programming.

Under certain conditions with high densities, the resulting linear program may be infeasible, in which case the ORCA model selects the "safest possible" velocity using a three-dimensional linear program.

In this work we use the RVO2 C++ library [12] as the implementation of the ORCA algorithm.

4 Results

We simulated the three test cases as described in Section 2 using the three modelling approaches presented in Section 3. For the Social Force model, we defined the desired speed v_α^0 according to [10] from a normal distribution with mean value $\mu = 1.34\,m/s$ and standard deviation $\sigma = 0.26$. All other parameters were set according to [13]. In order to have a maximum pedestrian speed in the Cellular Automaton model corresponding to the 95% percentile of this distribution we set the time step to 1/4.65s. To achieve variations in speed the probability for non-movement steps is set to 28%. Using this parameter set trajectories can be produced with an average velocity of 1.34 m/s. We implemented the long-range interactions between pedestrians by a continuous floor field. For the ORCA model the default parameters were used [5]. Note, that the parameter sets of the three models are left unchanged during all simulation runs.

In the test cases 6 and 12 the final goal is not visible from every point in the starting zone. Therefore an intermediate goal is placed manually to guarantee validity of all trajectories. The changeover to the final goal takes place when a pedestrian has approached to the intermediate goal nearer than one meter.

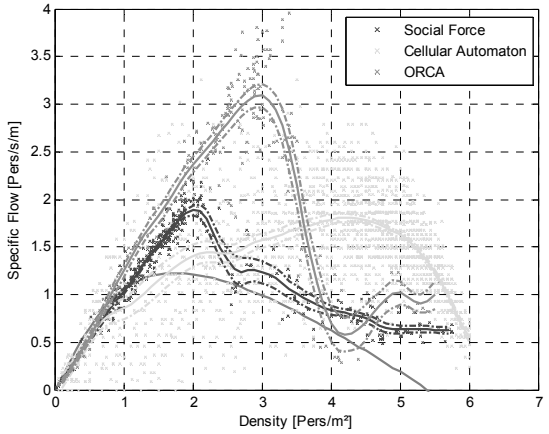
In this section we demonstrate and discuss the results of our model comparisons.

4.1 Results for Test Case 4

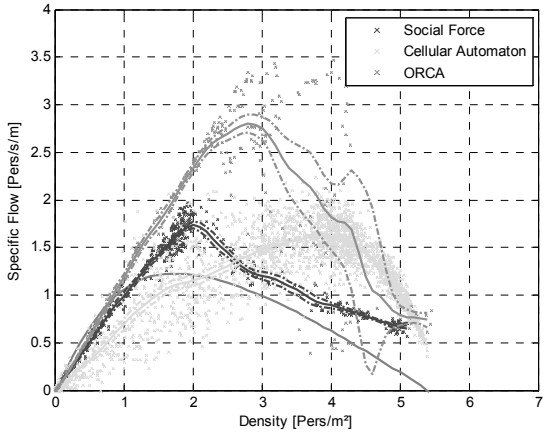
We performed five simulation runs for each average density ranging from 1 to 5 persons/m². All of the tested models were able to simulate pedestrian movement in the defined corridor setting as demanded in the description of the test case in Section 2.1. The resulting Fundamental Diagrams are shown in Figure 5.

Highly localized measurements maintain the homogeneity of density. Hence, we first used a measurement area of 2x2m for this test case (see Figure 5a). However, the results of the Cellular Automaton reveal unrealistic high deviations of flow: at higher densities the average flow is not decreasing as expected.

The implementation of the Cellular Automaton defines speed variations by a certain probability of keeping a position or moving to a neighbouring cell. While on a global perspective the average velocities can be derived correctly, it generates high errors for local measurements. This strong influence of the measurement method was already discussed in [14].



(a)



(b)

Figure 5. Fundamental Diagrams as result from the three tested models using a measurement area of (a) 2x2m and (b) 4x4m from Test Case 4. The red line corresponds to the fundamental diagram given in [10].

When switching to a larger measurement area of 4x4m (see Figure 5b) also higher heterogeneity in the local densities is introduced.

The effects from the smaller measurement area are reduced for the Cellular Automaton and the shape of the typical flow over density relationship is apparent. Also the results from the Social Force and the ORCA model reproduce a similar shape of the flow-density curve. Both models generate higher flow rates than given in [10] and in case of the ORCA model the maximum is shifted to a higher density.

4.2 Results for Test Case 6

In this test case, we have performed 10 simulation runs for each model. Our results confirm that all tested models are able to replicate movement around a corner without stepping through the walls. A qualitative evaluation of the simulation for walking around a corner is illustrated in Figure 6. It has to be noted that we defined a sub-goal, which is located one meter away from the inner corner on a line connecting the vertices of the inner and outer corner. Thus, pedestrians do not steer directly to the vertex of the inner corner. Figure 7 shows the empirical cumulative distribution function of the walking times from all simulation runs.

The resulting average and extreme values of the walking times from the 10 simulation runs are shown in Table 1. These results show that the three tested models simulate significant different walking times in this test case. The average walking times are more than twice as large for the Cellular Automaton than for the Social Force model. Pedestrians simulated with the Social Force model can move smoothly around the corner which results in the fastest average walking times. In contrast, the ORCA model creates congestions in the area of the corner which slow down the pedestrian flow. The pedestrians in the Cellular Automaton choose lower individual velocities than in the other two models as the densities in this test case are relatively low.

Figure 6. Simulation results from the three tested models at time t = 10s for Test Case 6.

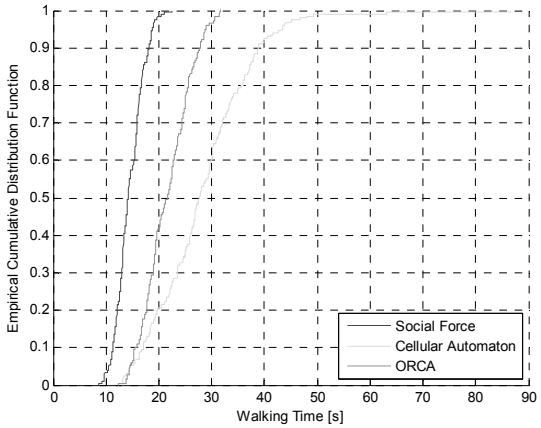


Figure 7. Empirical cumulative distribution of walking times from Test Case 6.

4.3 Results for Test Case 12

We have performed 10 simulation runs for each model in this test case. Again, we have defined a sub-goal at the centre of the entrance to the bottleneck in Room 1. The qualitative results shown in **Fehler! Verweisquelle konnte nicht gefunden werden.** reveal the differences in the pedestrian behaviour in conjunction with bottlenecks: the Social Force model creates congestions in front of the bottleneck in Room 1 which forms a half circle. At the beginning there is almost no congestion in front of the bottleneck in Room 2, but over time the density increases too. In contrast, the Cellular Automaton generates two walking lanes inside the corridor most of the time. As a consequence, the throughput is significantly higher than in the two other models. The ORCA model creates strong turbulences in the movement of pedestrians in front of the bottleneck in Room 1. This restricts the pedestrians from walking into the bottleneck and creates unrealistic high waiting times in front of the bottleneck in Room 1.

These qualitative observations are also confirmed by the empirical cumulative distribution function of the walking times shown in **Figure 9**. While the cumulative distribution of walking times for the Social Force model and the Cellular Automaton show a similar trend, these times are significantly longer for the ORCA model.

Figure 10 illustrates the densities which were measured for the three tested models in the two areas of both rooms. As expected, all models reveal higher densities in Room 1 for Area A. For Area B, the Social Force model and the Cellular Automaton reach densities of over 4 persons/m² while the ORCA model stays below this value.

In Room 2, the Social Force model and the Cellular Automaton show both higher densities in Area A which is directly in front of the bottleneck. In the surrounding area (i.e. Area B) these two models produce higher densities for short time periods only.

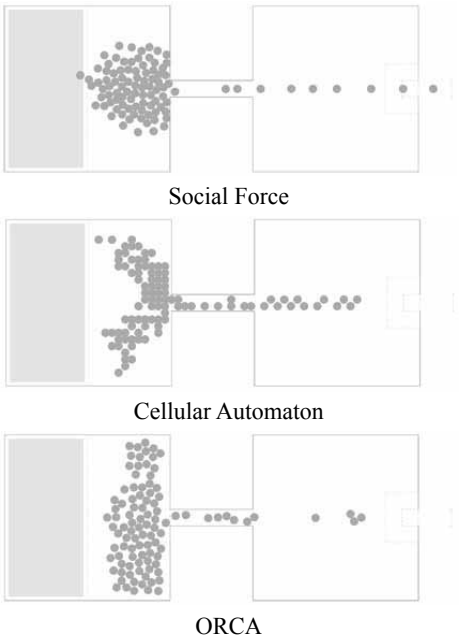


Figure 8. Simulation results from the three tested models at time t = 15s for Test Case 12.

In contrast, the ORCA model creates no congestions in neither of the two areas within Room 2. It can be seen that this is a result of the inflow restrictions into the bottleneck of Room 1. The original hypothesis of Test Case 12, namely, that congestion only appears at the exit of the first room and not at the exit of the second room is satisfied only by the ORCA model.

The resulting average and extreme values of the walking times from the 10 simulation runs are shown in Table 1. For the Cellular Automaton and the Social Force model, the average walking times are in the same range whereas the values for the ORCA model are twice as large as for the other two tested models.

	Average Walking Times (Min,Max) [s]	
	Test Case 6	Test Case 12
Social Force	19.1 (16.0,22.5)	185.0 (129.2,347.1)
Cellular Automaton	50.6 (38.9,86.7)	161.1 (121.9,269.5)
ORCA	29.8 (25.6,31.7)	394.7 (338.3,456.7)

Table 1. Walking times from Test Case 6 and Test Case 12.

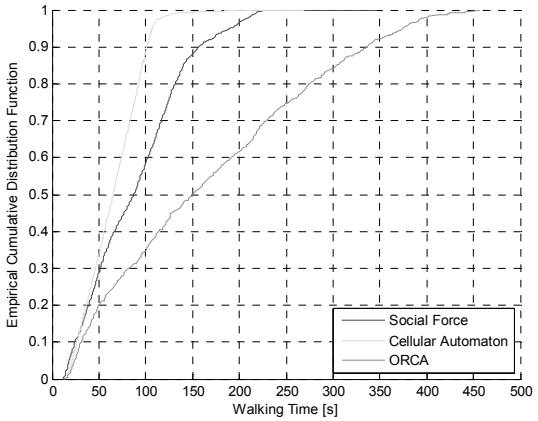


Figure 9. Empirical cumulative distributions of walking times from Test Case 12.

5 Summary

Based on the comparison of the three modelling approaches in Section 4 it is hard to infer systematic behaviour for each individual model.

For instance, the Cellular Automaton revealed the longest average walking time in Test Case 6. However, in Test Case 12, it had the least average walking time among the tested models. An explanation for this behaviour can be found in the Fundamental Diagram from Figure 5: compared to the other two models, the Cellular Automaton keeps a higher average flow rate at high densities and a lower average flow rate at low densities. A further reason are wall effects in Test Case 12 which keep the pedestrians in the Social Force and the ORCA model in single lanes walking through the corridor whereas in the Cellular Automaton model the pedestrians are walking in double lanes (see Figure 8).

The used implementation of the ORCA model shows strong turbulences in the pedestrian flow in front of bottlenecks (see [15]) which explains the long evacuation times in Test Case 12.

Furthermore, our results of Test Case 4 have underlined the importance of suitable methods for measuring and evaluating fundamental diagrams [14].

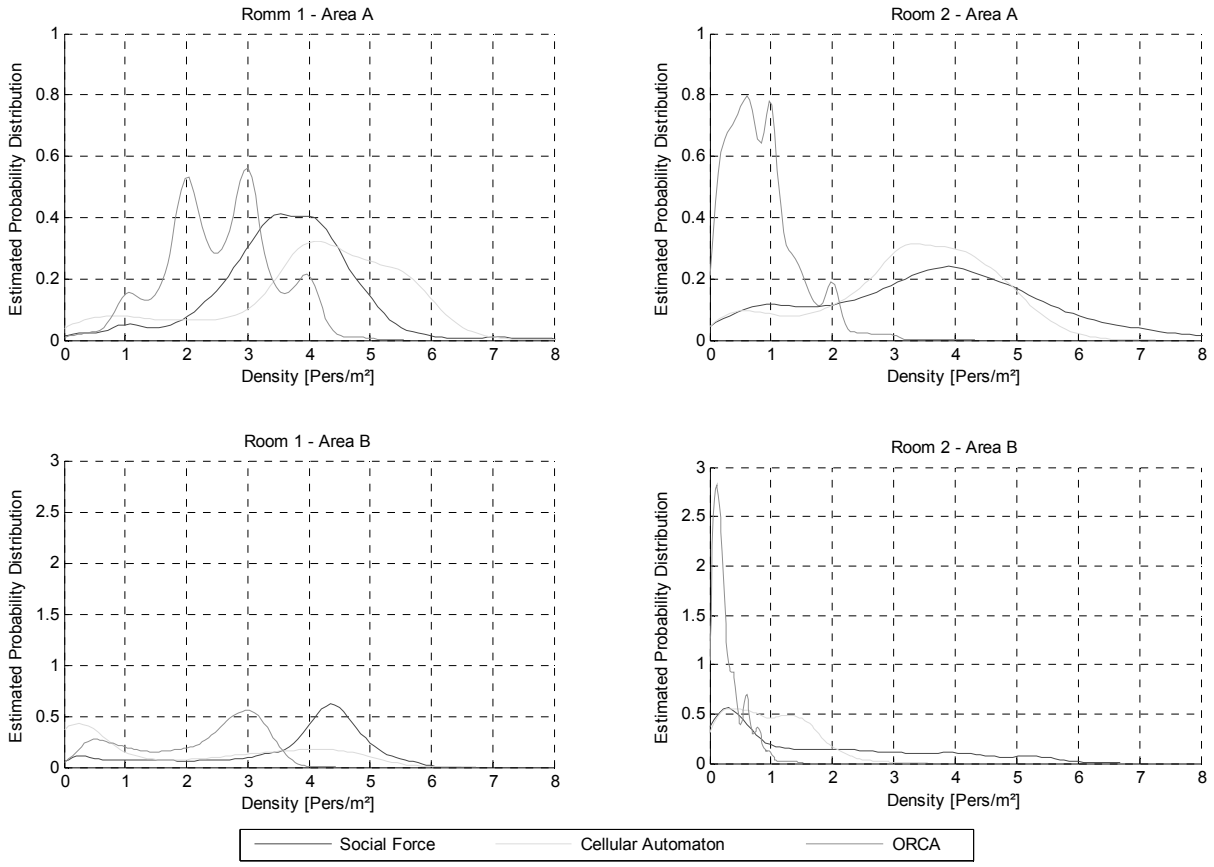


Figure 10. Densities measured for the three tested models from Test Case 12.

6 Conclusion and Perspectives

The presented work compared implementations of the Social Force model, Cellular Automaton and Optimal Reciprocal Collision Avoidance model based on three selected test cases from the RiMEA-Guideline. This comparison showed differing results of the three modelling approaches in all three test cases but did not provide identifiability of systematic behaviour for each individual model.

From our results of the test cases we cannot disprove any of the three tested models. An explanation for this is that the encoded criteria to successfully fulfil the tests are not strict (with the exceptions that pedestrians must reach their goal and must not walk through walls) and lack of quantifiable measures, such as minimum and maximum boundaries for average walking times in Test Case 6 and 12.

Currently, empirical data from experimental or real-world observations are not included in the RiMEA test cases. Since the effects described in the test cases are therefore rather based on assumptions or human observations, there is a strong need for empirical underpinning of these effects. This goes along with an extension of quantitative evaluation metrics for pedestrian movement characteristics beyond, for instance, the Fundamental Diagram.

In particular, three points need further discussion: 1) investigating which procedures are most suitable to measure and evaluate pedestrian movement characteristics, 2) including real world data into the test cases and 3) providing empirical data and well studied methods for model calibration. We understand this work as a further step towards making different microscopic pedestrian simulation models comparable and accelerating the development of benchmark and validity tests of such models.

References

[1] Rogsch C, Klüpfel H, Könnecke R, Winkens A. *RiMEA: A Way to Define a Standard for Evacuation Calculations*. Pedestrian and Evacuation Dynamics (PED2012), Springer International Publishing; 2014. p 455-467.

[2] Ronchi E, Kuligowski E, Reneke P, Peacock R, Nilsson D. *The Process of Verification and Validation of Building Fire Evacuation Models*. NIST Technical Note; 2013. p 1822.

[3] Helbing D, Johansson A. *Pedestrian, Crowd and Evacuation Dynamics*. Encyclopedia of Complexity and Systems Science 16(4); 2009: p 6476–6495.

[4] Burstedde C, Klauck A, Schadschneider A, Zittartz J. *Simulation of Pedestrian Dynamics using a 2-dimensional Cellular Automaton*. Physica A, vol 295; 2001. p 507–525.

[5] Van den Berg J, Guy SJ, Lin MC, Manocha D. *Reciprocal n-body collision avoidance*. In: The 14th International Symposium On Robotics Research (ISRR2011), Springer, vol 70; 2011. p 3–19.

[6] Kretz T. *Applications of the Dynamic Distance Potential Field Method*. Traffic and Granular Flow '09, Springer-Verlag; 2009.

[7] Davidich M, Köster G. *Towards automatic and robust adjustment of human behavioral parameters in a pedestrian stream model to measured data*. Safety Science. 2012; 50(5): p 1253-1260. ISSN 0925-7535.

[8] RiMEA. *Richtlinie für Mikroskopische Entfluchtungsanalysen*, Version: 2.2.1, Available online: www.rimea.de, 2009.

[9] Isenhour ML, Löhner R. *Verification of a Pedestrian Simulation Tool Using the NIST Recommended Test Cases*. Transportation Research Procedia. 2014; 2: p 237-245. ISSN 2352-1465, 2014.

[10] Weidmann U. *Transporttechnik der Fußgänger*. Schriftenreihe des Institut für Verkehrsplanung, Transporttechnik, Strassen- und Eisenbahnbau Nr. 90. Zürich; 1992. p 35–46.

[11] Helbing D, Molnár P. *Social Force Model for Pedestrian Dynamics*. Physical Review E. 1995; 51: p 4282–4286.

[12] RVO2 Library C++ v2.0.1 (released October 26, 2010), see http://gamma.cs.unc.edu/RVO2/

[13] Rudloff C, Matyus T, Seer S, Bauer D. *Can walking behavior be predicted? An analysis of the calibration and fit of pedestrian models*. In: Transportation Research Record, Volume 2264, ISSN: 0361-1981; 2011. p 101-109.

[14] Seyfried A, Boltes M, Kähler J, Klingsch W, Portz A, Rupprecht T, Schadschneider A, Steffen B. *Enhanced Empirical Data for the Fundamental Diagram and the Flow Through Bottlenecks*. Pedestrian and Evacuation Dynamics (PED2008), Springer Berlin Heidelberg; 2010. p 145-156.

[15] Berseth G, Kapadia M, Haworth B, Faloutsos P. *SteerFit: Automated Parameter Fitting for Steering Algorithms*. In: ACM SIGGRAPH/Eurographics Symposium on Computer Animation, ACM; 2014. p 10.

Different Methods analysing Convection-Diffusion

Stefanie Winkler[1*], Martin Bicher[1,2]

[1]Institute for Analysis and Scientific Computing, Vienna University of Technology, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria; *stefanie.winkler@tuwien.ac.at

[2]dwh GmbH, Neustiftgasse 95-97, 1070 Vienna, Austria

Simulation Notes Europe SNE 25(1), 2015, 43 - 48
DOI: 10.11128/sne.25.tn.10281
Received: November 8, 2014; Revised January 9, 2015;
Accepted: February 15, 2015;

Abstract. Many countries in this world have lack of drinking water. Austria has advantage of drinking water coming from the mountains. This article contains a study focusing on mathematical modelling using different methods for the analysis of groundwater pollution. The distribution of pollution follows the convection-diffusion equation. Therefore different methods ranging from analytical and numerical to alternative approaches dealing with random walk are compared. The analysis of the approaches is mostly done for one and two dimensional case.

Introduction

In order to analysis the pollution distribution in water of similar circumstances the mathematical equation describing this behaviour is a convection-diffusion equation. This equation can not only be used to analysis the behaviour of pollution. Also in biology, chemistry and other fields of study this equation is important. Regarding biology the equation can be used to predict the development of fur pattern for cats. In chemistry the mixture of different substances follows this equation. In the field of physical modelling and simulation this equation is often called heat equation because it describes the distribution of heat emanating from a source. Despite disciplines in natural sciences also the finance market uses this equation to foresee the behaviour of buyers of stocks. In general the convection- diffusion equation looks as follows:

$$\frac{\partial \mathrm{c}}{\partial \mathrm{t}} = \mathrm{D} \cdot \nabla^2 c - v \cdot \nabla c \tag{1}$$

Equation (1) is a partial differential equation of second order and contains two different variables D, v which can be time-dependent, position-dependent or simply constant. In the following we assume that all the variables are constant. The first term of this equation describes a regular distribution in every direction. It is similar to spreading of waves after throwing in a little stone into water. The variable v in the second term of (1) symbolises the velocity field of oriented movement. Assuming for example a river with a certain flux then the distribution would be influence by the velocity of the flux. This information will be transformed into the equation using the variable v. To sum it up, the convection-diffusion equation contains one part describing the chaotic movement in all directions and an oriented distribution depending on the circumstances. In the following a flux only in x-direction is assumed. This problem description will be analysed using three different approaches applied in one and two dimensions.

1 Analytical Solution

In this case, due to the used initial and boundary conditions, an analytical solution can be given. The initial condition describes a pollution sources which releases all the pollution at time $t = 0$ without injecting any further pollution. Both solutions, one- and two-dimensional, are used to validate the different methods.

One-dimensional. Using the regarded equation is given as follows

$$\frac{\partial \mathrm{c}}{\partial \mathrm{t}} = \mathrm{D} \cdot \frac{\partial^2 \mathrm{c}}{\partial \mathrm{x}^2} - v \cdot \frac{\partial \mathrm{c}}{\partial \mathrm{x}} \tag{2}$$

and has to fulfill the initial $c(x_0, 0) = \delta(x)$ and the boundary conditions $\lim_{x \to \pm\infty} c(x,t) = 0$. Using substitutions described in [1] the equation (2) can be written as

$$\begin{gathered} \tau = Dt, \qquad b = \frac{v}{D} \\ \mathrm{y} = \mathrm{x} - \mathrm{b}\tau, \qquad \mathrm{y}_0 = \mathrm{b}\tau_0 \\ \frac{\partial \mathrm{c}(\mathrm{y}, \tau)}{\partial \tau} = \frac{\partial^2 \mathrm{c}(\mathrm{y}, \tau)}{\partial \mathrm{y}^2} \end{gathered} \tag{3}$$

The resulting line in equation (3) can be multiplied by $e^{-p\tau}$. After integration with respect to τ one obtains an ordinary differential equation which can be solved using basic mathematical tools. A Laplace back transformation and backward substitution gives the solution of equation (2).

$$c(x,t) = \frac{1}{\sqrt{4\pi Dt}} e^{-\frac{(x-vt)^2}{4Dt}} \tag{4}$$

Two-dimensional. In the two dimensional case the obtained equation changes to

$$\frac{\partial c}{\partial t} = D \cdot \frac{\partial^2 c}{\partial x^2} + D \cdot \frac{\partial^2 c}{\partial y^2} - v \cdot \frac{\partial c}{\partial x} \tag{5}$$

Analogue to the one-dimensional case certain initial and boundary conditions are defined as follows

$$c(x_0, y_0, 0) = \delta(x)\delta(y)$$
$$\lim_{x,y\to\infty} c(x,y,0) = 0$$
$$\lim_{x,y\to-\infty} c(x,y,0) = 0$$

In order to solve equation (5) a specific form of the solution is assumed.

$$c(x,y,t) = g_1(x,x_0,t)g_2(y,y_0,t) \tag{6}$$

The functions g_1 and g_2 are solutions of the one-dimensional convection- diffusion equation with constant coefficients. Therefore g_1 and g_2 can be taken from the one-dimensional analytical solution (4).

$$g_1(x,x_0,t) = \frac{A_1}{2\sqrt{D\pi t}} e^{-\frac{(x-x_0-vt)^2}{4Dt}}$$
$$g_2(y,y_0,t) = \frac{A_2}{2\sqrt{D\pi t}} e^{-\frac{(y-y_0)^2}{4Dt}} \tag{7}$$

The source is located at the origin therefore the values $x_0 = 0$ and $y_0 = 0$ can be inserted. Additionally the integral over the whole domain has to be 1.

$$1 = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} c(x,y,t) =$$
$$= \int_{-\infty}^{\infty} g_1(x,0,t)dx \int_{-\infty}^{\infty} g_2(y,0,t)dy = A_1A_2 \tag{8}$$

This integration result leads to the analytical solution in two dimensions.

$$c(x,y,t) = \frac{1}{4Dt\pi} e^{-\frac{(x-vt)^2}{4Dt}} \tag{9}$$

2 Numerical Approximation

This section introduces two types of numerical approximations. On the one hand there is the finite difference method (FDM). In this approximation the derivative of the differential equation is approached by taking the difference quotient of the neighboring grid points. The method is easy to use but slightly weak concerning the accuracy. The second method is the finite element method (FEM) and is based on formulating variations of the differential equation. FEM determines approximated solutions consisting of piece wise defined polynomials on a fine resolution of the domain. The advantage of FEM is the suitability for any geometry.

2.1 Finite Difference Method

One-dimensional. Using finite differences to approximate the first and second derivatives the partial differential equation (2) transforms into an ordinary differential equation.

$$\frac{dc}{dt} = D \cdot \frac{c_{i+1} - 2c_i + c_{i-1}}{dx^2} - v \cdot \frac{c_i - c_{i-1}}{dx} \tag{10}$$

The time derivative can be replaced as follows

$$\frac{dc}{dt} = \frac{c^{k+1} - c^k}{\Delta t} \tag{11}$$

Using (11) equation (10) can also be written as a matrix product

$$\frac{c^{k+1} - c^k}{\Delta t} = S \cdot c^k \tag{12}$$

whereas c^k is the current concentration of pollution and c^{k+1} the concentration in the next time step. In order to determine c^{k+1} using the Explicit Euler equation (12) is rearranged.

$$c^{k+1} = (S \cdot \Delta t + I)c^k \tag{13}$$

It is well known that the Explicit Euler can be unstable using the wrong step size relation. Notation (12) can be also used to find the Implicit Euler formulation. The current concentration on the right hand side in equation (13) is replaced by the concentration of the future time step in order to obtain the implicit formulation.

$$c^{k+1} = (I - S \cdot \Delta t)^{-1} c^k \tag{13}$$

Two-dimensional. Regarding the problem formulation in two dimensions the finite difference method looks a little bit different. Due to the fact that an equidistant grid, $dx = dy$ is used the approximation can be given as follows

$$\frac{dc}{dt} = D \cdot \frac{c_{x+1,y} + c_{x-1,y} - 4c_{x,y} + c_{x,y+1} + c_{x,y-1}}{dx^2} - v \cdot \frac{c_{x,y} - c_{x-1,y}}{dx} \tag{14}$$

In contrary to the two-dimensional case the matrix nota-

tion is not as easy as in one dimension.

$$c_{x,y}^{k+1} = c_{x,y}^{k} + h\frac{dc}{dt} \tag{15}$$

Therefore only the Explicit Euler method is implemented as shown in (15).

2.2 Finite Element Method

The finite element method was only realized for the convection-diffusion equation in one dimension.

$$\begin{gathered}\frac{\partial c}{\partial t} - D\frac{\partial^2 c}{\partial t^2} + v\frac{\partial c}{\partial x} = 0 \text{ in } \Omega \\ c = 0 \text{ on } \partial\Omega\end{gathered} \tag{16}$$

First of all the weak solution of (16) is formalized using a test function of the according Sobolev space $\phi \epsilon H_0^1$.

$$\int_\Omega \frac{\partial c}{\partial t}\phi \, d\Omega + \int_\Omega (D\,\nabla c\,\nabla\phi + v\nabla c\,\phi) = 0 \tag{17}$$

The formulation of the Galerkin approximation is necessary to formulate the solution equation of the finite element method.

$$c^n(x) = \sum_{j=1}^{n} c_j\varphi_j(x) + c_0(x) \tag{18}$$

The unknown variables c_j in equation (18) have to be determined. Using linear basis functions called 'hat-functions' for φ a linear system of n equations with n unknowns, called the Galerkin formulation, results [3].

$$\begin{gathered}\sum_{j=1}^{n_e} \frac{\partial c_j}{\partial t} \int_{\Omega^{e_k}} \varphi_i\,\varphi_j \, d\Omega + \\ \sum_{j=1}^{n_e} c_j \int_{\Omega^{e_k}} (D\,\nabla\varphi_j\,\nabla\varphi_i + \nabla\varphi_j\,\varphi_i)\, d\Omega = 0\end{gathered} \tag{19}$$

In equation (19) n_e is the number of elements in every finite element and Ω^{e_k} is the domain of element e_k. Equation (19) can also be written in a short form.

$$\begin{gathered}\dot{c}\cdot M + c\cdot S = 0 \\ m_{ij} = \int_{\Omega^{e_k}} \varphi_i\,\varphi_j \, d\Omega \\ s_{ij} = \int_{\Omega^{e_k}} (D\,\nabla\varphi_j\,\nabla\varphi_i + \nabla\varphi_j\,\varphi_i)\, d\Omega\end{gathered} \tag{20}$$

The matrices of (20) are called mass matrix M and stiffness matrix S. Considering the mentioned 'hatfunctions' it is clear, that only a few of the possible integrals are not equal zero.

Those basis functions which correspond to the corner points of the element will lead to non trivial results. Because the element i is connected to $i-1$ and $i+1$ the profile of the matrices is a band matrix with width three.

$$\begin{gathered}M\frac{c^{k+1} - c^k}{\Delta t} + \theta S c^{k+1} + (1-\theta) S c^k = 0 \\ 0 \le \theta \le 1\end{gathered} \tag{21}$$

Equation (22) is called θ-method and will be used to present implicit and explicit methods for solving (21). The most common values for q are:

- $\theta = 0$, Eplicit Euler
- $\theta = 1$, Implicit Euler
- $\theta = \frac{1}{2}$, Implicit Heun

Using this method the Explicit and Implicit Euler algorithm can be given.

$$\begin{aligned}c^{k+1} &= M^{-1}(M - \Delta t S)c^k \\ c^{k+1} &= (M + \Delta t S)^{-1} M\, c^k\end{aligned} \tag{22}$$

3 Random Walk

An alternative method for simulating transport is the so-called random walk. This approach is contrary to the numerical solutions. The focus changes from a macroscopic view to the simulation of microscopic behavior of diffusion by analyzing movements of single particles.

3.1 Intuitive Approach

The intuitive approach describes a model which uses no grid or collision rules. It is implemented again for both dimensions.

One-dimensional. At the beginning t = 0 all the particles are placed in the origin presenting the source of pollution. The pollution injection happens only at t = 0. The simulation focuses on the convection and diffusion behaviour of these initial particles. In this approach the movement of particles is described by:

$$\begin{gathered}p_{new} = p_{old} \\ r = X \cdot \Delta x\end{gathered} \tag{23}$$

The particle motion in (23) consists of three parts. In order to get the new position p_{new} at time $t + \Delta t$ these three components are summed up. The variable p_{old} stands for the position at time t. The velocity field v is multiplied by the step size. The variable r describes the diffusive movement of a particle for one time step and is added to the former particle position p_{old}.

The second equation in (23) defines the movement r in particular. It consists of the step size in space Δx and a normally distributed random variable X with mean zero and unit variance. In every time step the new position of every particle is calculated with equation (23).

The simulation ends when the chosen simulation time tend is reached.

Two-dimensional. For expansion in a two-dimensional domain the movement has to be defined in a different way. There is no initial velocity but there is an initial direction of every particle d_0. The diffusive transport is realized by using a normally distributed random variable X and a uniformly distributed random number U. X is used to generate a random length and U chooses a coincidental direction.

$$r = X \cdot \Delta x \qquad \alpha = U \cdot 2\pi$$
$$d_0 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad d_{n+1} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \cdot d_n \tag{24}$$

In (24) r stands for the distance the particle moves in a certain time step. The influence of this parameter is similar to the diffusion coefficient. X is the mentioned normally distributed random variable and Δx describes the step size in space. The second equation of (24) sets the direction for the particle's next move. The initial direction d_0 is only necessary for the recursive definition. During simulation the direction of the last movement is used to calculate the next one. The convection is realized by a shift in flow direction along x. The final formulation of the random walk movement can be given as follows

$$p_{new} = p_{old} + d \cdot r + v\,\Delta t \tag{25}$$

3.2 Gaussian Approach

This approach shows the connection between a random walk approach and the analytical solution.

One-dimensional. The analytical solution of the convection-diffusion equation (2) is used to define the particle movement. Considering the probability density function of a normal or Gaussian distribution

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{26}$$

At the beginning t = 0 all the particles are placed in the origin presenting the source of pollution. The pollution injection happens only at t = 0. The simulation focuses on the convection and diffusion behaviour of these initial particles. In this approach the movement of particles is described by:

$$\begin{aligned} \mathrm{p_{new}} &= \mathrm{p_{old}} \\ r &= X \cdot \Delta x \end{aligned} \tag{27}$$

the formal equivalence to the analytical solution (4) is obvious. The parameters used in (27) stand for the mean value m and the standard deviation s which characterize the position and the width of the Gaussian bell curve in a unique way. Therefore the according parameters in (4) can be read out. [4]

$$\mu = v \cdot t \qquad \sigma^2 = 2 \cdot Dt \tag{28}$$

Due to the properties and meaning of the parameters in (28) the height and width of the concentration peak depending on time is given. The corresponding particle movement using (29) can be formulated as follows

$$p_{new} = p_{old} + v\,\Delta t + \sqrt{2D\Delta t} \cdot X \tag{29}$$

The variable X stands for a normally distributed random number with mean zero and unit variance as in the intuitive approach. X is newly generated in every step for each particle. Identifiable by the velocity v the second term stands for the convective motion. This term is equal to the term of the intuitive approach. The radical term describes the diffusive motion and is based on the standard derivation.

Two-dimensional. In order to enlarge this approach in two dimensions the movement along y- direction has to be added. For an expansion in a two-dimensional domain the y-component of the movement has to be defined. Due to the fact that there is no flux the new particle position can be calculated using

$$\begin{aligned} p_x^{new} &= p_x^{old} + v\,\Delta t + \sqrt{2D\Delta t} \cdot X_x \\ p_y^{new} &= p_y^{old} + \sqrt{2D\Delta t} \cdot X_y \end{aligned} \tag{30}$$

The variables X_x and X_y stand for independent normally distributed random numbers which are newly generated in every step for each particle. The term $v\Delta t$ describes the convective transport. Due to the fact that the diffusion coefficient is equal for the x- and y-direction the diffusive movement $\sqrt{2D\Delta t}$ in the random walk definition (30) is the same.

4 Results

In the following section the analytical solutions in both dimensions are compared to the various approaches. The different concentration errors are discussed. In general the parameter setting is: diffusion coefficient $D = 0.02$ and velocity $v = 0.02$.

The step sizes Δt and Δx are variable. The regarded simulation time varies between $t_{end} = 250$ and $t_{end} = 500$.

4.1 Analytical vs. Finite Difference Method Results

First of all the numerical solutions are considered.

One-dimensional. In the plot below in Figure 1 the red curve is the analytical solution and the blue line sketches the numerical approximation using the Implicit Euler algorithm.

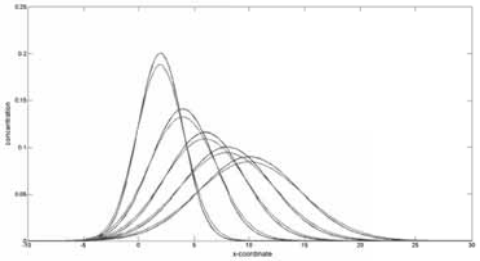


Figure 1. Comparison of the analytical solution and FDMusing matrix notation.

The results in Table 1 show the instability of the Explicit Euler method. The Implicit Euler algorithm is not only ultra-stable but also faster and more exact than the Explicit Euler. The approximation using finite differences is well-fitting.

Δt	Δx	Explicit Euler $\|.\|_\infty$	Explicit Euler $\|.\|_1$	Implicit Euler $\|.\|_\infty$	Implicit Euler $\|.\|_1$
1	1	0.016	$4.231E^{-4}$	0.016	$4.753E^{-4}$
1	$\frac{1}{2}$	0.009	$1.404E^{-4}$	0.010	$1.600E^{-4}$
$\frac{1}{2}$	$\frac{1}{4}$	0.005	$0.831E^{-5}$	0.005	$7.323E^{-5}$
$\frac{1}{2}$	$\frac{1}{16}$	NaN	NaN	0.002	$3.531E^{-5}$

Table 1. Error values of FEM using Explicit and Implicit Euler.

Two-dimensional. The results regarding the two-dimensional implementation show a similar behaviour. In the following the error values are studied in detail.

Also in the two-dimensional case the Explicit Euler works not for all parameter choices. The error values are again quite good. The finite difference method of the two-dimensional domain approximates the convection-diffusion equation in an appropriate way.

Δt	Δx	Explicit Euler $\|.\|_\infty$	Explicit Euler $\|.\|_1$
1	1	0.027	$1.5624E^{-4}$
1	$\frac{1}{2}$	0.017	$3.779E^{-5}$
$\frac{1}{2}$	$\frac{1}{4}$	$9.148E^{-4}$	$1.464E^{-5}$
$\frac{1}{4}$	$\frac{1}{8}$	$.E^{119}$	$.E^{120}$

Table 2. The error values for FDM are shown.

4.2 Analytical vs. Finite Element Method Results

The accuracy of the finite element method is better than of the finite difference method.

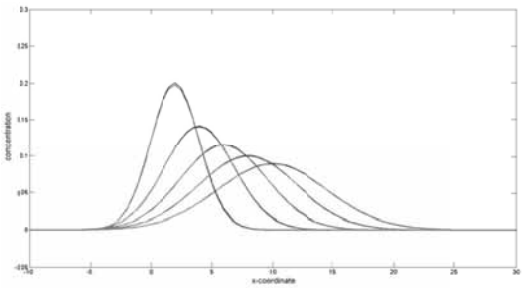


Figure 3. The error for the Implicit Euler algorithm of the FEM is shown.

In Figure 3 above the upper plot shows the analytical solution as well as the finite element method using Implicit Euler. It is hard to distinguish the different curves.

Δt	Δx	Explicit Euler $\|.\|_\infty$	Explicit Euler $\|.\|_1$	Implicit Euler $\|.\|_\infty$	Implicit Euler $\|.\|_1$
1	1	$7.18E^{-4}$	$3.16E^{-5}$	$9.95E^{-4}$	$3.03E^{-5}$
1	$\frac{1}{2}$	$6.23E^{-4}$	$8.60E^{-5}$	$6.09E^{-4}$	$8.54E^{-5}$
$\frac{1}{2}$	$\frac{1}{4}$	$3.13E^{-4}$	$1.02E^{-4}$	$2.74E^{-4}$	$1.01E^{-4}$
$\frac{1}{4}$	$\frac{1}{8}$	NaN	NaN	$2.49E^{-4}$	$1.05E^{-4}$

Table 3. Depending on the used FEM error values are shown.

The instability of the Implicit Euler is shown in the last row of table 3. In general the error results are smaller compared to the results of the finite difference method in one dimension. The finite element method approximates the convection-diffusion equation better than the finite difference method.

4.3 Analytical vs. Stochastic Results

The accuracy of the random walk approaches is discussed in the following paragraph.

One-dimensional. In the plot below in Figure 1 the red curve is the analytical solution and the blue line sketches the numerical approximation using the Implicit Euler algorithm.

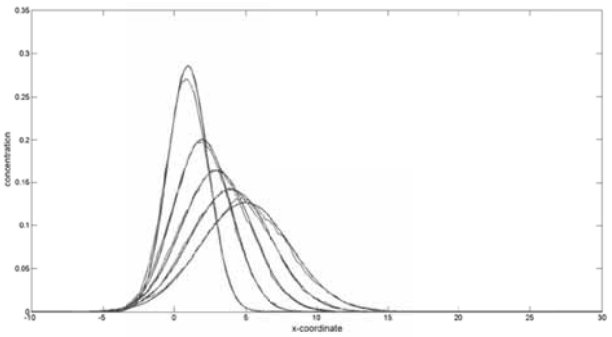


Figure 4. Results of stochastic based random walk are shown.

The graphic in Figure 4 show the Gaussian random walk approach coloured in red and the analytical solution in blue. In the numerical comparisons the simulation time is $t_{end} = 500s$. Due to long execution times for the particle movement this parameter is reduced to $t_{end} = 250s$. The diffusion coefficient is usually set to $D = 0.02$ but modifies if the intuitive approach is used.

		Gaussian	Random Walk
Δt	Δx	$\|.\|_\infty$	$\|.\|_1$
1	$\frac{1}{5}$	0.012	$8.948E^{-7}$
1	$\frac{1}{10}$	0.008	$9.707E^{-7}$
$\frac{1}{2}$	$\frac{1}{5}$	0.007	$8.948E^{-7}$
$\frac{1}{2}$	$\frac{1}{10}$	0.010	$9.707E^{-7}$

Table 4. Comparison of random walk analytical solution.

The Table 4 shows all the error results of the parameter study comparing the analytical solution and the random walk. The diffusion coefficient for the Gaussian-based algorithm is set to $D = 0.02$. Regarding simulation of the convection-diffusion equation, the implementation of the Gaussian-based random walk fits better than the intuitive approach. The number of particles is 6000.

Two-dimensional. In order to compare the analytical solution to a random walk approach the results have to be adapted. In the random walk the output describes the smoothed amount of particles in every cell. Due to the initial Dirac-function the integral at the beginning has value one. The area of the random walk domain is discretizised. Therefore the output has to be divided not only by the number of particles but also by the area of the cells used for the flattening. Table 5 shows the approximation results. The parameter r describes the used radius for the flattening. If the spatial step size is decreasing a greater radius r can be used. If r is chosen too big compared to Δx the result loses the shape of a bell curve. Compared to the results of the numerical simulation the random walk approach leads to greater error values. The number of particles is 4000.

				Implicit	Euler
Δt	Δx	r	N	$\|.\|_\infty$	$\|.\|_1$
1	1	3	4000	$3.395E^{-3}$	$6.349E^{-4}$
1	$\frac{1}{2}$	8	4000	$5.033E^{-3}$	$3.737E^{-5}$
$\frac{1}{2}$	$\frac{1}{4}$	15	4000	$4.526E^{-3}$	$1.005E^{-4}$
$\frac{1}{2}$	$\frac{1}{8}$	20	4000	$2.801E^{-3}$	$2.206E^{-3}$
1	$\frac{1}{4}$	20	8000	$6.764E^{-3}$	$1.826E^{-4}$

Table 5. Comparison of random walk analytical solution.

5 Conclusion

In general the finite element method approximates the convection-diffusion equation the best. Of course the very best solution is the analytical one. In spite of it all random walk approaches are quite good approximations of the convection-diffusion equation.

References

[1] Schulten K, Kosztin I. Lectures in Theoretical Biophysics, University of Illinois at Urbana, USA; 2000.

[2] Zoppou C,Knight JH. Analytical solution of a spatially variable coefficient advection diffusion equation in up to three dimensions, *Applied Mathematical Modelling*; 1999. p 667-685.

[3] Segal A. Finite element methods for the incompressible Navier-Stokes equations, Delft University of Technology, Netherlands; 2012.

[4] Salamon P, Fernandez-Garcia D, Gomez-Hernandez JJ. A review and numerical assessment of the random walk particle tracking method, *Journal of Contaminant Hydrology*; 2006. p 277-305.

Modelling and Simulation of a SIR-type Epidemic with Cellular Automata and Ordinary Differential Equations – Definition ARGESIM Benchmark C17R

Florian Miksch[1*], C. Haim[2], Günter Schneckenreither[12]

[1]dwh simulation services, Neustiftgasse 57-59, 1070 Vienna, Austria; **florian.miksch@dwh.at*

[2]Inst. of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Haupstraße 8-10, 1040 Vienna, Austria

Simulation Notes Europe SNE 25(1), 2015, 49 - 54
DOI: 10.11128/sne.25.bn17r.10283
Received: October 20, 2014; Revised February 10, 2015;
Accepted: February 15, 2015;

Abstract. This Comparison investigates a classical population model for the spread of infection diseases (SIR ordinary differential equations model by Kermack and McKendrick) and an inhomogeneous spatial approach using cellular automata. An identification of parameters based on an abstract time discrete conceptual model is presented. The tasks of this comparison include the validation and analysis of this identification, an investigation on the impact of different spatial dynamics in the cellular automaton modelling approach and simulation scenarios for confining epidemic outbreaks that involve state-dependent interventions.

Introduction

This comparison is a revision of the original ARGESIM Comparison 17 [1] and is targeted at the identification of the classical SIR-type differential equations model by Kermack and McKendrick [2] (which is a cumulative population model) with a microscopic individual based cellular automaton modelling approach [3].

For the purpose of a systematic identification of both approaches, a virtual individual based time-discrete population system with contact-induced SIR-characteristic spread of an infectious disease is presented. Based thereon the differential equations and cellular automaton models are derived and identified in an analytical fashion. Furthermore intervention scenarios for confining epidemic outbreaks are discussed.

1 System Definition

Let N be the number of individuals of the population. The population can neither be joined nor left by individuals, which means that N is a constant number. Each individual shall be in one of the states susceptible, infected or recovered. The system evolves by discrete steps of one time unit and the spread of the disease is characterised by contacts between individuals, transmission of the disease and recoveries.

Parameter	Description
S_0	initial number of susceptible
I_0	initial number of infected
R_0	initial number of recovered
C	contacts
α	infection probability
β	recovery probability

Table 1. System parameters.

Each individual is assumed to have an average of C contacts per time step; these contacts always happen between two random individuals. Since the discrete time steps are atomic by definition, the order of contacts is irrelevant. However, in order to ensure that susceptible individuals cannot get infected and infect others simultaneously, the infection-states of the individuals change after all contacts have been processed according to the following paradigms:

- When a susceptible individual gets into contact with an infected individual, the susceptible individual becomes infected with probability α. This probability applies for each contact separately.
- Infected individuals recover at the end of each time unit with probability β.
- Recovered individuals always remain recovered.

Interventions. In order to confine an epidemic, interventions might be applied. We define two different types of strategies ('soft' and 'hard') that can be applied when a certain critical threshold of infected individuals is reached or exceeded. The threshold is defined relative to the whole population as $f_T N$.

As a 'soft' strategy, the system parameters C, α or β are decreased to $f_S \cdot C$ (or $f_S \cdot \alpha$ or $f_S \cdot \beta$) over a period of time Δt. $0 \leq f_S \leq 1$ is called the reduction parameter. This can be either a linear decrease in the form $f(x) = k \cdot x$ or a smooth step in the form $f(x) = 3 \cdot x^2 - 2 \cdot x^3$. Let $r(t)$ be the function that describes the decrease from 1 to f_S, so that it can be multiplied with C, α or β, and let t_T be the time when the threshold is reached. For a linear decrease

$$r(t) = \left(1 - \frac{t - t_T}{\Delta t}\right) + f_s \frac{t - t_T}{\Delta t} \quad (1)$$

and for a smooth step it can be

$$r(t) = 1 - \left(3\left(\frac{t - t_T}{\Delta t}\right)^2 - 2\left(\frac{t - t_T}{\Delta t}\right)^3\right) \cdot (1 - f_s). \quad (2)$$

If $\Delta t = 0$, then the change is a discontinuous step. Figure 1 illustrates the idea and expected outcome of a 'soft' strategy.

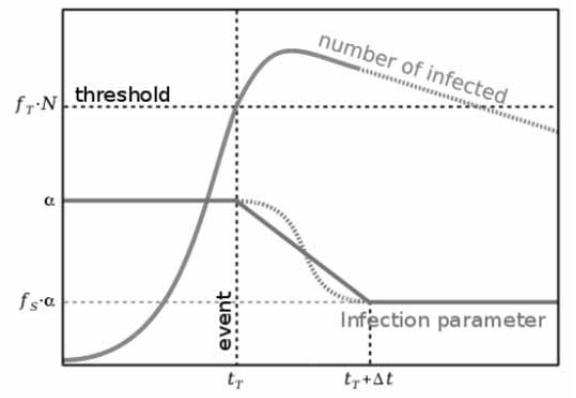


Figure 1. Illustration of a 'soft' intervention. Once the number of infected reaches a critical threshold, the infection parameter decreases over a certain period of time.

'Hard' strategies involve the individuals directly. Representing a quarantine or vaccination strategy, susceptible or infected individuals, respectively, can become recovered. Such a 'hard' strategy is defined by choosing a fraction f_H of susceptible or infected individuals and immediately changing their state to 'recovered' when the threshold of infected individuals is reached. The individuals are chosen randomly among all possible individuals with the respective state, since further distinction is not possible with this system definition.

Table 2 lists all parameters that are relevant for interventions.

It could be that the threshold is reached more than once. This happens, for example, when the number of infected is growing, then it is reduced by an intervention but still keeps growing. The intervention strategy can be applied in two ways: either only once when the treshold is reached for the first time or every time it is reached.

Parameter	Description
f_T	Fraction that defines the threshold
f_S	Reduction parameter of a soft intervention
Δt	Duration of a soft intervention
f_H	Fraction parameter of a hard intervention

Table 2. Parameter of hard and soft interventions.

2 Differential Equations Model

The differential equation, which models the defined system, corresponds to the classical SIR epidemic model which was proposed by W. O. Kermack and A. G. McKendrick in 1926 [2]. $S'(t)$, $I'(t)$ and $R'(t)$ represent the change of susceptible, infected and resistant individuals. The amount of susceptible individuals that become infected is described as $\gamma S(t) I(t)$, where γ is referred to as infection rate. The amount of infected individuals that become resistant is described as $\delta I(t)$, and δ is named recovery rate. Since the number of individuals in our system shall be constant, these growth terms yield the following system of ordinary differential equations (ODE) shown in (3).

$$\begin{aligned} S'(t) &= -\gamma \cdot S(t) \cdot I(t) \\ I'(t) &= \gamma \cdot S(t) \cdot I(t) - \delta \cdot I(t) \\ R'(t) &= \delta \cdot I(t) \end{aligned} \quad (3)$$

Before setting the ODE parameters, one needs to deal with another concern. An ODE $A'(t) = -\xi A(t)$ with $0 \leq \xi \leq 1$ represents a system where A is continuously decreased. However, ξ does not represent the amount of decrease within one time unit. If A should be decrease to $\hat{\xi} \cdot A$ within one time unit, then ξ needs to be set as

$$\xi = \ln(1 - \hat{\xi}). \quad (4)$$

This is based on the fact that the general solution of the ODE is $A(t) = A(0) \cdot e^{-\xi t}$. The condition that A should be decreased leads to the equation $A(t+1)=(1-\hat{\xi}) \cdot A(t)$, and further results in the formula above.

Identification of the infection term $\gamma \cdot S(t) \cdot I(t)$ takes a look at a single individual, which has C contacts per time unit in average. Among the contacts, it has $C \cdot \frac{I}{N}$ contacts with infected individuals. Each contact causes a transmission with probability α. The transmissions are statistically independent events. Hence, the infection probability per time unit is computed as the probability to get infected at least once, which is represented by the formula $1-(1-\alpha)^{C \cdot \frac{I}{N}}$. Considering Equation (4), the infections in the ODE are represented by $-\ln\left(1-1+(1-\alpha)^{C \cdot \frac{I}{N}}\right) \cdot S$, which can be rewritten as $-I \cdot S \cdot \frac{C}{N} \cdot \ln(1-\alpha)$. Hence, γ is identified with $-\frac{C}{N} \cdot \ln(1-\alpha)$.

Since the recovery rate δ determines the fraction of infected individuals that recover during one time unit, δ calculates as $-\ln(1-\beta)$.

The identified parameters are summarised in Table 3.

Parameter	Identification
$\boldsymbol{S(0)}$	S_0
$\boldsymbol{I(0)}$	I_0
$\boldsymbol{R(0)}$	R_0
$\boldsymbol{\gamma}$	$-\frac{C}{N} \cdot \ln(1-\alpha)$
$\boldsymbol{\delta}$	$-\ln(1-\beta)$

Table 3. Parameter identification of the differential equation model.

Interventions. In a 'soft' intervention strategy, the parameter γ or δ in the ODE system needs to switch to a time dependent function $\hat{\gamma}(t)$ or $\hat{\delta}(t)$ when the threshold $f_T N$ is reached at time $t = t_T$.

$\hat{\gamma}(t)$ or $\hat{\delta}(t)$ calculates as the term in Table 3 where the desired parameter C, α or β is replaced by $C \cdot r(t)$ or $\alpha \cdot r(t)$ or $\beta \cdot r(t)$, and $r(t)$ corresponds to the function in Equation (1) or (2).

At time $t = t_T + \Delta t$, the ODE system switches back to Equation (3) where γ or δ is replaced by $f_S\gamma$ or $f_S\delta$.

In a 'hard' intervention strategy the ODE system abruptly changes $S(t)$ or $I(t)$ when the threshold is reached. This can be achieved using the delta distribution $D(x)$ where $D(0) = 1$ and $D(x) = 0$ for $x \neq 0$

For example, quarantining the fraction f_H of infeceted individuals when the threshold is reached, then the ODE can be rewritten as Equation (5).

$$\begin{aligned} S'(t) &= -\gamma \cdot S(t) \cdot I(t) \\ I'(t) &= \gamma \cdot S(t) \cdot I(t) \\ &-\delta \cdot I(t) - D(I - I_T) \cdot f_H \cdot I(t) \\ R'(t) &= \delta \cdot I(t) + D(I - I_T) \cdot I_Q \end{aligned} \quad (5)$$

3 Cellular Automaton Model

In the context of modelling and simulation Cellular Automata (CA) can be seen as a time- and space-discrete modelling approach. A CA consists of cells which are arranged on a regular grid and can hold different states [4]. Lattice Gas Cellular Automata (LGCA) are an extension of the concept of CA where particles move around these cells [5]. Especially with a hexagonal lattice, LGCA are used for simulating the movement of gas particles or fluids.

We will additionally allow particles to take one of the states susceptible, infected or recovered [1, 3, 6] in order to simulate the spatial spread of a SIR-type disease.

Accordingly, we assume that our cells are arranged on a two-dimensional hexagonal grid structure and represent a spatial segment. Each cell can hold at most six individuals. Each individual is in one of the three states susceptible, infected or recovered. Contacts happen pairwise between all individuals which are located in the same cell at the same time. To simulate a mixture of the individuals, they move around the cells in random directions (diffusion) or as defined by the FHP-I collision rules [5]:

- The position of an individual within a cell defines its moving direction (Figure 2).
- After the movement phase a collision phase (Figure 3) takes place. The FHP-I variant only defines special two and three particle collisions. All other collisions happen without any change of moving direction. When two individuals collide as in Figure 3, they are reflected clockwise or counter clockwise with probability 0.5. When three particles collide as pictured in Figure 3, then they are reflected clockwise.

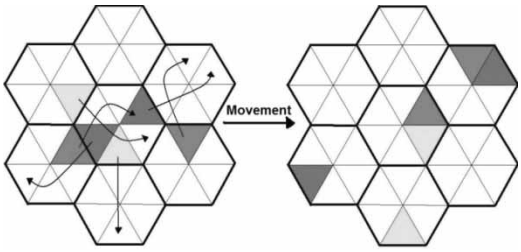


Figure 2. Schematic visualization of LGCA movement rules.

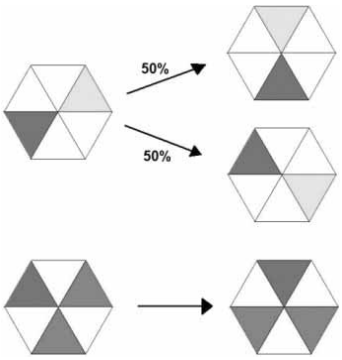


Figure 3. FHP-I collision rules.

When a susceptible individual meets an infected individual within a cell, it shall become infected with probability φAn infected individual recovers with probability ϕ.

The size of the LGCA plays an important role because it affects the density of particles and thus the number of contacts. For the sake of simplicity, we use a grid with $width = length = n$ and accordingly n^2 cells with six places each. Table 4 shows the parameters of the model with appropriate parameterisation. For a given number of individuals, the number of contacts depends on the size n of the LGCA.

The correct identification for n is crucial, but follows a simple calculation: Assuming a uniform distribution of the individuals, each of the six slots of a cell is occupied with the same probability. For a given individual there are 5 unoccupied slots in the same cell. Accordingly $N-1$ remaining individuals occupy $6n^2-1$ remaining slots and the individual has an expected number of

$$C = 5 \cdot \frac{N-1}{6n^2-1} \tag{6}$$

contacts within this cell. Adjusting n, which has to be an integer, to meet a given number of contacts leads to the identification in Table 4.

Parameter	Identification
$\boldsymbol{S(0)}$	S_0
$\boldsymbol{I(0)}$	I_0
$\boldsymbol{R(0)}$	R_0
$\boldsymbol{\varphi}$	α
$\boldsymbol{\phi}$	β
$\boldsymbol{n}$	$\left\lfloor \sqrt{\frac{5(N-1)+C}{6C}} \right\rfloor$

Table 4. Parameter identification of the cellular automaton model.

Interventions. Also for the CA approach intervention scenarios can be simulated. If the number of infected particles in the CA reaches the threshold, one of the strategies described in Section 1 can be applied.

Applying a 'soft' strategy is easy for α and β. Then, φ or ϕ are multiplied with $r(t)$ in Equation (1) or (2) so that the $f_S\varphi$ or $f_S\phi$ are reached after Δt. Reducing C causes problems because it requires a change of n. First, changing n is very inaccurate and second, there is no instruction on how to enlarge or shrink the space in respect to the individuals that have positions on the grid. Thus, if a 'soft' strategy for C is desired, it should be performed very carefully.

'Hard' intervention strategies can be directly applied as stated in the system definition. Then the threshold is reached, a desired number of susceptible or infected individuals are randomly chosen, and immediately become recovered.

In contrast to the ODE model, individuals in the CA are distinguished by their spatial location on the lattice. Presumably, the selection of particular individuals/particles for changing their state can make a crucial difference. For testing purposes, it seems reasonable to deliberately violate the system definition and choose individuals with respect to their location.

4 Analytical Comparison

There is a strong analytical relation between the ODE approach and the CA model. For simplicity the parameter identification presented in this section neglects interventions.

Infections. The following calculation aims to estimate the number of new infections in a time step in the LGCA. Consider a susceptible individual in a cell (only susceptible individuals can get infected). Then there are altogether $6n^2-1$ remaining slots in the LGCA, 5 remaining slots in the cell and I infected individuals. Define the probability of i slots in the cell being occupied by infected individuals as q_i. Under the assumption that the individuals are uniformly distributed, the number of infected individuals in this cell is distributed according to a hypergeometric distribution. The probabilities calculate as choosing i out of I infected individuals on 5 out of $6n^2-1$ places:

$$q_i = \frac{\binom{5}{i}\binom{(6n^2-1)-5}{I-i}}{\binom{6n^2-1}{I}}, \quad i = 0 \dots 5 \tag{7}$$

The expected value E of this hypergeometric distribution is

$$E = \sum_{i=0}^{5} q_i i = I\frac{5}{6n^2-1}. \tag{8}$$

Using the identification in (6), the expected value can be written as

$$E = I\frac{C}{N-1}. \tag{9}$$

With these preparations the actual infection probability of a susceptible individual can be calculated. If the cell is occupied by i infected individuals the probability for an infection of the susceptible individual is $1-(1-\alpha)^i$. Hence the expected probability for an infection is $\sum_{i=0}^{5} q_i(1-(1-\alpha)^i)$. Considering the first two terms of the Taylor series expansion at $\alpha = 0$ and the identification in (9) leads to the following approximation for this probability.

$$\begin{aligned}\sum_{i=0}^{5} q_i(1-(1-\alpha)^i) &\approx \sum_{i=0}^{5} q_i i\alpha = \\ &= \alpha\sum_{i=0}^{5} q_i i \;\; = \alpha E = \;\; \alpha I\frac{C}{N-1}\end{aligned} \tag{10}$$

Multiplying (10) with the total number of susceptible individuals leads to $S\alpha I\frac{C}{N-1}$ as an approximation for the expected total number of new infections for one time unit in the LGCA for small values of α.

The term in the ODE for infections of one time unit per susceptible is computed in section 2 as $1-(1-\alpha)^{\frac{C\cdot I}{N}}$. In the term for the CA, $\frac{I}{N-1}$ can be approximated with $\frac{I}{N}$. Natural limitations are $\frac{I}{N} \le 1$ and $C \le 5$. For small α and the natural limitations, $1-(1-\alpha)^{\frac{C\cdot I}{N}}$ is an approximation of $\alpha\frac{C\cdot I}{N}$.

Recoveries. An infected individual in the LGCA recovers during one time unit with probability β, hence the expected amount of infected individuals who regenerate in one time unit is βI. The same factor also occurs in the differential equation (3) and Table 3 as $-\ln(1-\beta)$.

5 Tasks

In order to validate the analytical findings from Section 4 in an experimental fashion, both model approaches must be implemented in a simulation environment or as stand-alone programs. The primary output and point of comparison for both approaches is the evolution of the numbers $S(t)$, $I(t)$ and $R(t)$.

5.1 Task 1 – Model Comparison

We use the parameters from Table 5 as a starting point and perform a parameter analysis for I_0, α and β while keeping the overall population count ($N = 10\,000$) constant. For the CA approach, FHP-I collision rules and a uniform distribution of the individuals as initial condition seem legit.

a) Is it possible to identify parameter regions with similar behaviour in both modelling approaches? It is very likely that the reasons for qualitative and quantitative differences lie in the fact that the identification of the infection parameter is based on the assumption that the population is always uniformly distributed on the lattice and secondly that a Taylor approximation was used in (10).

b) From Equation (10) we can see that the average probability of a susceptible individual to get infected depends on the contact rate C and the infection probability α. We can analyse the trade-off between those two parameters by performing a parameter variation of α and C leaving the product αC constant. The dynamics of the CA imply different effects of variations in α and C respectively. Note that, in particular, changing the contact rate C also involves changing the size n of the lattice and that the contact rate for the six-particle LGCA is limited.

Parameter	Value
S_0	9 500
I_0	500
R_0	0
C	4
α	0.1
β	0.1

Table 5. Parameter set for Task 1.

5.2 Task 2 – Interventions

It is rather clear that different intervention strategies deliver different reactions of the system. Additionally for both modelling approaches the same intervention strategy can have different effects (compare Task 1).

a) Based on a parameter set with similar behaviour in both modelling approaches (Task 1) choose a threshold f_T for the number of infected and compare different intervention strategies in both modelling approaches.
b) What is the advantage of the spatial LGCA approach in combination with the 'hard' strategies? For example we can assume that vaccinating or quarantining individuals at the interface between regions with high and low infections reduces the spread of the disease. On the other hand, 'hard' interventions in the center of infected areas may have hardly any effect on the epidemic.

5.3 Task 3 – Spatial Inhomogeneity

A constantly homogeneous mixture of the population in the CA model can be achieved by introducing a third type of 'movement' rules: instead of moving to an adjacent cell, the particles jump to an arbitrary cell on the lattice ('random movement'). Based on the findings from Task 1 we can choose a parameter setting for which the ODE approach and the original CA model deliver different qualitative behaviour. We can however postulate that the CA model with this new type of movement rules behaves identical to the ODE model. How can this be explained?

References

[1] Hötzendorfer H, Popper N, Breitenecker F. *Temporal and Spatial Evolution of a SIR-type Epidemic – ARGES-IM Comparison C17 – Definition*, Simulation News Europe; 2004.

[2] Kermack WO,McKendrick AG. *A Contribution to the Mathematical Theory of Epidemics*, Proc. R. Soc. Math. Phys. Eng. Sci.. 1927; 115(772): p 700–721.

[3] Fukś H, Lawniczak AT. *Individual-based lattice model for spatial spread of epidemics.* Discrete Dynamics in Nature and Society. 2001;6: p 191–200.

[4] Wolfram S. *A new kind of science*. Champaign, IL: Wolfram Media; 2002.

[5] Wolf-Gladrow DA. *Lattice-Gas Cellular Automata and Lattice Boltzmann Models: An Introduction*, 1st ed. Berlin: Springer; 2000.

[6] Yakowitz S, Gani J, Hayes R. *Cellular automaton modeling of epidemics*, Appl. Math. Comput.. 1990; 40(1): p 41–54.

A SimEvents/Simulink -based Solution to ARGESIM Benchmark C8 'Canal and Lock System'

Dominik Brunmeir[1*], Matthias Rößler[1,2]

[1]Department of Analysis and Scientific Computing, Vienna University of Technology, Wiedner Haupstraße 8–10, 1040 Vienna, Austria; [*]*dominik.brunmeir@gmail.com*

[2]dwh GmbH, Neustiftgase 57–59, A-1070 Vienna, Austria

SNE Simulation Notes Europe SNE 25(1), 2015, 55-58
DOI: 10.11128/sne.25.bn08.102 85
Received: July 10, 2014; Revised February 10, 2015;
Accepted: February 15, 2015;

Abstract. This paper discusses a model for a system of two canals and a lock, with transiting ships and their average transit time. Different policies for the lock are investigated and variance reduction experiments are conducted.

Introduction

The Model, as described in ARGESIM Comparison C 8, was implimented in Simulink, and is an example of modelling complex logic. The statistical analysis was done in Gnumeric, an open-source Spreadsheet application. The use of Simulink made it easy to monitor the various outputs.

1 Description of the System

The system consists of two canals and a central lock. The ships pass through in batches in a single direction. So it is necessary to implement a logical system that switches the direction of transversal. We have to apply rules to make sure that after a certain amount of ships, the other direction gets access to the lock as well, even if there are still ships waiting. This maximum amount is called Eastmax or Westmax, depending on the direction.

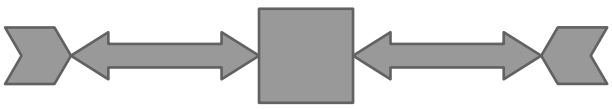


Figure 1: Simple schematics of the Canal and Lock System.

1.1 Canal

A ship always has to pass through two canals to get through the system. If it is an eastbound ship it will first move 14 minutes throught the west canal, pass the lock in a variable amount of time and then pass through the east canal in 18 minutes. A westbound ship will do the same in reverse. Because the canals are too narrow to fit two ships side by side, the whole system can only be traversed in one direction simultaneously.

1.2 Lock

The Lock raises or lowers the waterlevel to the other canal, when a ship is passing through the system. It can hold only one ship simultaneously and needs twelve minutes to raise or lower the water. A ship takes five minutes to enter and five minutes to leave the lock. The time a ship spends in the lock may vary, because the lock is able to raise or lower the waterlevel to the approaching ship as soon as a ship enters the canal preceeding the lock. The minimal time spent is therefore 22 minutes, and the maximum time 34 minutes.

1.3 Direction

As mentioned, the canals are too narrow to let ships move through simultaneously in both directions, so rules are necessary to allow ships to transit with no deadlock as two ships with opposite directions approach one canal. If there are no ships in the system, when a new ship arrives, it enters the system immediately and starts a cycle. If another ship arrives, while the previous ship is in transit, and if it travels in the same direction, it will also enter the system and is added to the cycle unless Eastmax or Westmax ships have entered the system in a single cycle. If Eastmax or Westmax is reached, the following ship is denied access and the direction will be reversed after the last barge of a cycle has passed

through the system. In case there is no ship waiting for entrance in the opposite direction, a new cycle in the same direction will begin.

2 Description of the Model

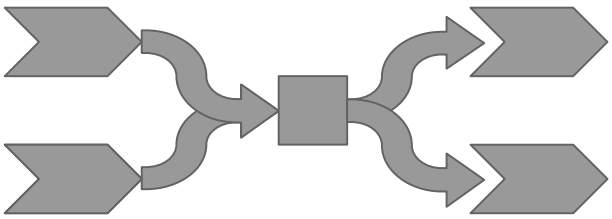


Figure 2: Simple schematics of the Model.

The model is implemented in Matlab Simulink with the help of Matlab SimEvent blocks. The ships are represented by entities which move through several gates and servers. For the implementation we changed the basic structure to create a more linear flow of all entities. Instead of two bidirectional canals, we built the model with two unidirectional canals, with variable lengths of transit time dependant on the origin of the ship entities. Therefore *east* and *west* is a bit of a misnomer, but is still applied to the variable and function names to allow an easy way to match the corresponding building blocks of the model to the system.

Overview. At first glance we see in 3 the flow of the model going from left to right, with the cycler subsystem on the top controling the flow of the system. We will now detail the various important subsystems further.

The Waiting Line. The ship entities are generated by a time based entity generator, which gets its intervals between ship generation, from the start script. Those entities continue to server blocks, that represents the waiting line before the canals. Given that the procedure of passing through the first canal is equal for eastbound or westbound ships and only differs in the amount of time spent when passing through the canals, we used an input switch block to open the first canal block for the appropriate entity. This switch block is controlled by the cycler subsystem.

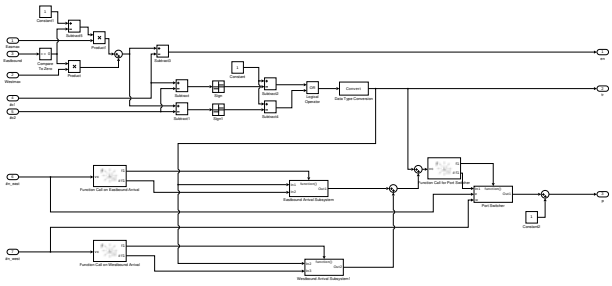


Figure 4: Cycler Subsystem

The Cycler Subsystem. A subsystem called cycler implements the logic behing the switching of the direction and opening the first canal. It also secures that only Eastmax or Westmax number of ships may pass through in one batch. Instead of reversing the direction it flips a switch block to allow the right amount of barges, moving in the right direction to pass through the system. The upper part, as seen in 4, decides if Eastmax or Westmax is reached, while the bottom part uses function calls on arrival of new entities in the waiting server blocks to decide wether the ship entity may continue into the system.

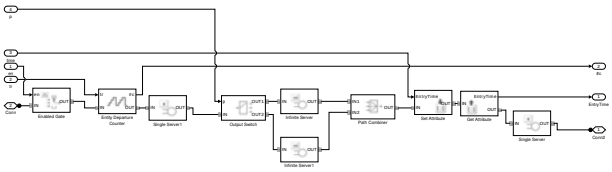


Figure 5: First Canal Subsystem

The First Canal. In the first canal we have a gate and a series of server blocks that represent the amount of time that is necessary to pass through the canal, depending on the direction. The opening of the gate is controlled by the cycler subsystem.

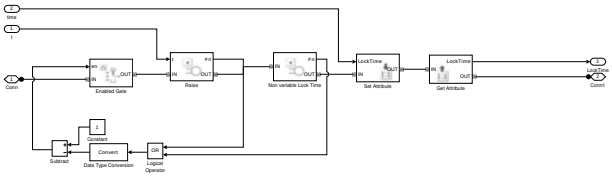


Figure 6: Lock Subsystem

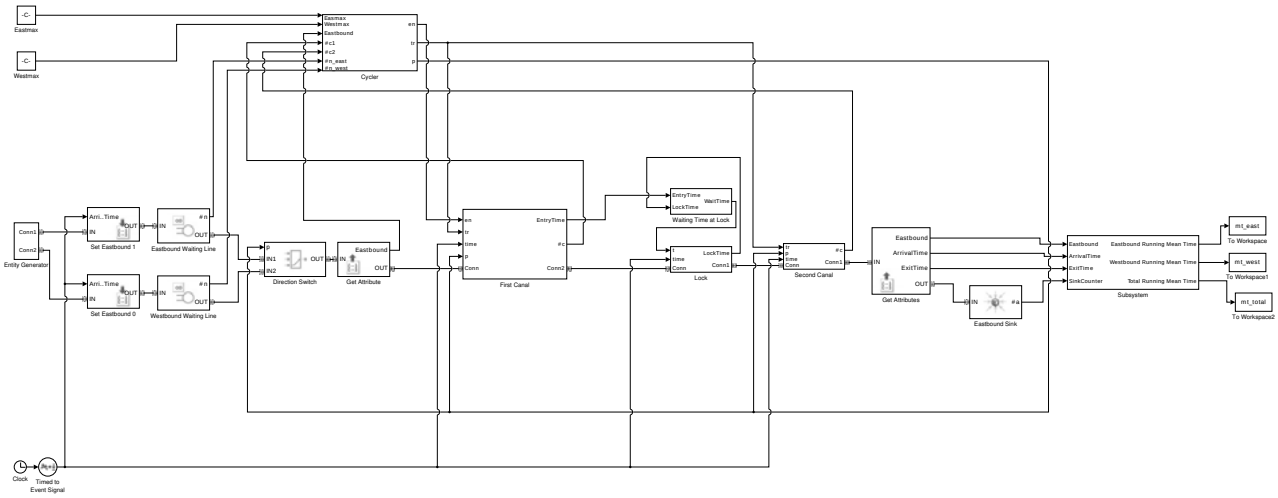


Figure 3: Overview of the Model

The Lock. The next subsystem is the lock itself, which also consists of gates and servers. We seperated the time spent in the lock into two parts, the non-variable part and the variable part. We block the entrance into the lock with an enabled gate that only allows entities in, if the system is empty. Then follows a server block, that represents the variable part of the time spent in the lock. To calculate the service time in the server block we pass along the time the previous ship left the lock.

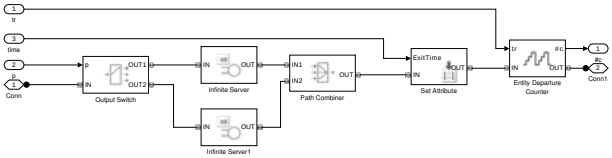


Figure 7: Second Canal Subsystem

The Second Canal. Now follows the second canal which is just a series of servers, with a fixed service time, dependant on the direction of the ship entities.

Exit. Before leaving the system, the transit time of the entity, as well as a moving mean are calculated. This is necessary for the Model validation and the variance reduction experiments.

2.1 Model validation

The validation of the Model was done with given datasets, which, in contrast to the later variance reduction experiments, also included cases with multiple simultaneously arriving ships.

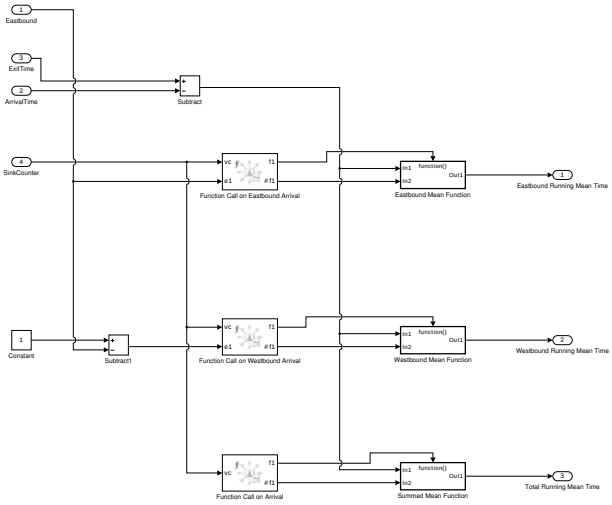


Figure 8: Exit Subsystem

3 Variance reduction experiments

For the variance reduction experiments, the arrival times of the ships were pre-calculated. While the interval between the arrival of two ships is exponentially distributed with a mean of 75, the first ship arrives at an uniformly distributed time, with a mean time of 15 minutes for the eastbound, and ten minutes for the westbound ship. For this, the pseudo random number

generator of matlab was used. The end time was set to 14400 minutes to simulate a ten day period and 100 independent replications were conducted. The maximum number of ships for a full cycle was set to five in both directions. For better insight, the whole simulation was repeated two more times. All the following numbers were calculated with Gnumeric 1.12.17.

	Mean	**90% CI**	σ^2
Run 1	511,17	36,17	219,88
Run 2	472,25	37,49	227,93
Run 3	520,96	37,74	229,47

Table 1: Three independent runs of the simulation

The use of the antithetic random variates variance-reduction methology brought a significant contraction of the confidence interval and σ^2.

	Mean	**90% CI**	σ^2
Run 1	547,88	30,19	129,77
Run 2	536,81	31,62	135,93
Run 3	553,61	31,52	135,50

Table 2: Three independent runs of the simulation using the ARV variance-reduction method.

The next experiment compared the mean transit times, when the cyclelength is set to six, to the cyclelenght of five in both directions. We considered the null hypothesis that the expected transit time for five-ship-cycles is less than or equal than six-ship-cycles. 50 replications for the first case and 50 for the second case were done and the mean times subtracted. A negative number means that the six-ship-cycle took longer than the 5-ship-cycle.

	Difference	**90% CI**	σ^2
Run 1	72,66	73,85	317,48
Run 2	137,99	76,92	330,69
Run 3	39,40	67,86	291,73

Table 3: The difference between Eastmax and Westmax set to six or five.

Based on the result of these runs we cannot reject the null hypothesis. But with the use of common random numbers (CRNs), we get a different conclusion.

	Difference	**90% CI**	σ^2
Run 1	67,50	7,05	30,33
Run 2	69,52	8,52	36,64
Run 3	59,20	8,34	35,87

Table 4: The difference between Eastmax and Westmax set to six or five with CRNs.

As we can see, the intervals are strictly positive and thus we can safely reject the null hypothesis. The reduction of the confidence interval is not surprising, but still remarkable.

References

[1] Schriber T J. Comparison 8: Canal-and-Lock System. *SNE*. 1995; 6(1): 29–31.

SNE Simulation News

EUROSIM Data and Quick Info

UNIVERSITY of OULU

EUROSIM 2016

9th EUROSIM Congress on Modelling and Simulation

City of Oulu, Finland, September 16-20, 2016

www.eurosim.info

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. **SNE** is published in a printed version (Print ISSN 2305-9974) and in an online version (Online ISSN 2306-0271). With **Online SNE** the publisher **ARGESIM** follows the **Open Access** strategy for basic **SNE** contributions. Since 2012 **Online SNE** contributions are identified by DOI 10.11128/sne.xx.nnnnn. for better web availability and indexing.

Print SNE, high-resolution **Online SNE**, and additional **SNE** contributions are available via membership in a **EUROSIM** society.

This *EUROSIM Data & Quick Info* compiles data from EUROSIM societies and groups: addresses, web links, officers of societies with function and email, to be published regularly in **SNE** issues.

SNE Editorial Office /ARGESIM

→ *www.sne-journal.org, www.eurosim.info*
office@sne-journal.org (info, news)
eic@sne-journal.orgt Felix Breitenecker (publications)

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.

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 advancement of modelling and simulation in industry, research, and development. → *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 fourteen *Full Members* and three *Observer Members*:

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

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

President	Esko Juuso (SIMS) *esko.juuso@oulu.fi*
Past President	Khalid Al.Begain (UKSIM) *kbegain@glam.ac.uk*
Secretary	Borut Zupančič (SLOSIM) *borut.zupancic@fe.uni-lj.si*
Treasurer	Felix Breitenecker (ASIM) *felix.breitenecker@tuwien.ac.at*
SNE Repres.	Felix Breitenecker *felix.breitenecker@tuwien.ac.at*

SNE – Simulation Notes Europe. **SNE** is a scientific journal with reviewed contributions as well as a membership newsletter for **EUROSIM** with information from the societies in the *News Section*. **EUROSIM** societies are offered to distribute to their members the journal **SNE** as official membership journal. **SNE** Publishers are **EUROSIM**, ARGESIM and ASIM.

Editor-in-chief	Felix Breitenecker *felix.breitenecker@tuwien.ac.at*

→ *www.sne-journal.org,*

✉ *office@sne-journal.org*

EUROSIM Congress. **EUROSIM** is running the triennial conference series **EUROSIM** Congress. The congress is organised by one of the **EUROSIM** societies.

EUROSIM 2016 will be organised by SIMS in Oulu, Finland, September 16-20, 2016.

Chairs / Team EUROSIM 2016

Esko Juuso EUROSIM President, *esko.juuso@oulu.fi*
Erik Dahlquist SIMS President, *erik.dahlquist@mdh.se*
Kauko Leiviskä EUROSIM 2016 Chair,
kauko.leiviska@oulu.fi

→ *www.eurosim.info*

✉ *office@automaatioseura.fi*

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 700 individual members, and 30 institutional or industrial members.

→ *www.asim-gi.org* with members' area

info@asim-gi.org, admin@asim-gi.org

✉ AS IM – Inst. f. Analysis and Scientific Computing
Vienna University of Technology
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, *pawel@mb.hs-wismar.de*
Secretary	Ch. Deatcu, *christina.deatcu@hs-wismar.de*
Treasurer	Anna Mathe, *anna.mathe@tuwien.ac.at*
Membership Affairs	S. Wenzel, *s.wenzel@uni-kassel.de*
	W. Maurer, *werner.maurer@zhwin.ch*
	Ch. Deatcu, *christina.deatcu@hs-wismar.de*
	F. Breitenecker, *felix.breitenecker@tuwien.ac.at*
Universities / Research Inst.	S. Wenzel, *s.wenzel@uni-kassel.de*
	W. Wiechert, *W.Wiechert@fz-juelich.de*
	J. Haase, *Joachim.Haase@eas.iis.fraunhofer.de*
	Katharina Nöh, *k.noeh@fz-juelich.de*
Industry	S. Wenzel, *s.wenzel@uni-kassel.de*
	K. Panreck, *Klaus.Panreck@hella.com*
Conferences	Klaus Panreck *Klaus.Panreck@hella.com*
	J. Wittmann, *wittmann@htw-berlin.de*
Publications	Th. Pawletta, *pawel@mb.hs-wismar.de*
	Christina Deatcu, *christina.deatcu@hs-wismar.de*
	F. Breitenecker, *felix.breitenecker@tuwien.ac.at*
Repr. EUROSIM	F. Breitenecker, *felix.breitenecker@tuwien.ac.at*
	N. Popper, *niki.popper@drahtwarenhandlung.at*
Education / Teaching	A. Körner, *andreas.koerner@tuwien.ac.at*
	N. Popper, *niki.popper@drahtwarenhandlung.at*
	Katharina Nöh, *k.noeh@fz-juelich.de*
International Affairs	A. Körner, *andreas.koerner@tuwien.ac.at*
	O. Rose, *Oliver.Rose@tu-dresden.de*
Editorial Board SNE	T. Pawletta, *pawel@mb.hs-wismar.de*
	Ch. Deatcu, *christina.deatcu@hs-wismar.de*
Web EUROSIM	Anna Mathe, *anna.mathe@tuwien.ac.at*

Last data update December 2013

ASIM Working Committee. **ASIM**, part of GI - Gesellschaft für Informatik, is organised in Working Committees, dealing with applications and comprehensive subjects in modelling and simulation:

ASIM Working Committee	
GMMS	Methods in Modelling and Simulation Th. Pawletta, *pawel@mb.hs-wismar.de*
SUG	Simulation in Environmental Systems Wittmann, *wittmann@informatik.uni-hamburg.de*
STS	Simulation of Technical Systems H.T.Mammen, *Heinz-Theo.Mammen@hella.com*
SPL	Simulation in Production and Logistics Sigrid Wenzel, *s.wenzel@uni-kassel.de*
EDU	Simulation in Education/Education in Simulation N. Popper, *niki.popper@dwh.at* A. Körner, *andreas.koerner@tuwien.ac.at*
	Working Groups for Simulation in Business Administration, in Traffic Systems, for Standardisation, for Validation, etc.

CEA-SMSG – Spanish Modelling and Simulation Group

CEA is the Spanish Society on Automation and Control In order to improve the efficiency and to deep into the different fields of automation, the association is divided into thematic groups, one of them is named 'Modelling and Simulation', constituting the group.

→ *www.cea-ifac.es/wwwgrupos/simulacion*

→ *simulacion@cea-ifac.es*

✉ CEA-SMSG / María Jesús de la Fuente,
System Engineering and AutomaticControl department,
University of Valladolid,
Real de Burgos s/n., 47011 Valladolid, SPAIN

CAE - SMSG Officers	
President	M. À. Piera Eroles, *MiquelAngel.Piera@uab.es*
Vice president	Emilio Jiminez, *emilio.jiminez@unirioja.es*
Repr. EUROSIM	Emilio Jiminez, *emilio.jiminez@unirioja.es*
Edit. Board SNE	Emilio Jiminez, *emilio.jiminez@unirioja.es*
Web EUROSIM	Mercedes Peres, *mercedes.perez@unirioja.es*

Last data update December2013

CROSSIM – Croatian Society for Simulation Modelling

CROSSIM-*Croatian Society for Simulation Modelling* was founded in 1992 as a non-profit society with the goal to promote knowledge and use of simulation methods and techniques and development of education. CROSSIM is a full member of **EUROSIM** since 1997.

→ *www.eurosim.info*

vdusak@foi.hr

✉ CROSSIM / Vesna Dušak
Faculty of Organization and
Informatics Varaždin, University of Zagreb
Pavlinska 2, HR-42000 Varaždin, Croatia

CROSSIM Officers	
President	Vesna Dušak, *vdusak@foi.hr*
Vice president	Jadranka Božikov, *jbozikov@snz.hr*
Secretary	Vesna Bosilj-Vukšić, *vbosilj@efzg.hr*
Executive board members	Vlatko Čerić, *vceric@efzg.hr*
	Tarzan Legović, *legovic@irb.hr*
Repr. EUROSIM	Jadranka Božikov, *jbozikov@snz.hr*
Edit. Board SNE	Vesna Dušak, *vdusak@foi.hr*
Web EUROSIM	Jadranka Bozikov, *jbozikov@snz.hr*

Last data update December2012

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*). The main objectives of the society 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*
Treasurer	Evžen Kindler, *ekindler@centrum.cz*
Scientific Secr.	A. Kavička, *Antonin.Kavicka@upce.cz*
Repr. EUROSIM	Miroslav Šnorek, *snorek@fel.cvut.cz*
Deputy	Mikuláš Alexík, *alexik@frtk.fri.utc.sk*
Edit. Board SNE	Mikuláš Alexík, *alexik@frtk.fri.utc.sk*
Web EUROSIM	Petr Peringer, peringer@fit.vutbr.cz

Last data update December2012

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.eurosim.info

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	A. Heemink, *a.w.heemink@its.tudelft.nl*
Vice president	W. Smit, *smitnet@wxs.nl*
Treasurer	W. Smit, *smitnet@wxs.nl*
Secretary	W. Smit, *smitnet@wxs.nl*
Repr. EUROSIM	A. Heemink, *a.w.heemink@its.tudelft.nl*
Deputy	W. Smit, *smitnet@wxs.nl*
Edit. Board SNE	A. Heemink, *a.w.heemink@its.tudelft.nl*

Last data update April 2006

FRANCOSIM – Société Francophone de Simulation

FRANCOSIM was founded in 1991 and aims to the promotion of simulation and research, in industry and academic fields. Francosim operates two poles.

- Pole Modelling and simulation of discrete event systems. Pole Contact: *Henri Pierreval, pierreva@imfa.fr*
- Pole Modelling and simulation of continuous systems. Pole Contact: *Yskandar Hamam*, *y.hamam@esiee.fr*

→ *www.eurosim.info*

y.hamam@esiee.fr

FRANC OSIM / Yskandar Hamam
Groupe ESIEE, Cité Descartes,
BP 99, 2 Bd. Blaise Pascal,
93162 Noisy le Grand CEDEX, France

FRANCOSIM Officers	
President	Karim Djouani, *djouani@u-pec.fr*
Treasurer	François Rocaries, *f.rocaries@esiee.fr*
Repr. EUROSIM	Karim Djouani, *djouani@u-pec.fr*
Edit. Board SNE	Karim Djouani, *djouani@u-pec.fr*

Last data update December2012

HSS – Hungarian Simulation Society

The Hungarian Member Society of EUROSIM was established in 1981 as an association promoting the exchange of information within the community of people involved in research, development, application and education of simulation in Hungary and also contributing to the enhancement of exchanging information between the Hungarian simulation community and the simulation communities abroad. HSS deals with the organization of lectures, exhibitions, demonstrations, and conferences.

→ *www.eurosim.info*

javor@eik.bme.hu

HSS / András Jávor,
Budapest Univ. of Technology and Economics,
Sztoczek u. 4, 1111 Budapest, Hungary

HSS Officers	
President	András Jávor, *javor@eik.bme.hu*
Vice president	Gábor Szűcs, *szucs@itm.bme.hu*
Secretary	Ágnes Vigh, *vigh@itm.bme.hu*
Repr. EUROSIM	András Jávor, *javor@eik.bme.hu*
Deputy	Gábor Szűcs, *szucs@itm.bme.hu*
Edit. Board SNE	András Jávor, *javor@eik.bme.hu*
Web EUROSIM	Gábor Szűcs, *szucs@itm.bme.hu*

Last data update March 2008

ISCS – Italian Society for Computer Simulation

The Italian Society for Computer Simulation (ISCS) is a scientific non-profit association of members from industry, university, education and several public and research institutions with common interest in all fields of computer simulation.

→ *www.eurosim.info*

Mario.savastano@uniina.at

ISCS / Mario Savastano,
c/o CNR - IRSIP,
Via Claudio 21, 80125 Napoli, Italy

ISCS Officers	
President	M. Savastano, *mario.savastano@unina.it*
Vice president	F. Maceri, *Franco.Maceri@uniroma2.it*
Repr. EUROSIM	F. Maceri, *Franco.Maceri@uniroma2.it*
Secretary	Paola Provenzano, *paola.provenzano@uniroma2.it*
Edit. Board SNE	M. Savastano, *mario.savastano@unina.it*

Last data update December2010

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, organization of courses and stages in companies to apply the simulation to real problems.

→ *www.liophant.org*

info@liophant.org

LIOPHANT Simulation, c/o Agostino G. Bruzzone,
DIME, University of Genoa, Polo Savonese,
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 Masseimassei@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 December2013

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.

→ *briedis.itl.rtu.lv/imb/*

merkur@itl.rtu.lv

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

LSS Officers	
President	Yuri Merkuryev, *merkur@itl.rtu.lv*
Secretary	Artis Teilans, *Artis.Teilans@exigenservices.com*
Repr. EUROSIM	Yuri Merkuryev, *merkur@itl.rtu.lv*
Deputy	Artis Teilans, *Artis.Teilans@exigenservices.com*
Edit. Board SNE	Yuri Merkuryev, *merkur@itl.rtu.lv*
Web EUROSIM	Oksana Sosho, *oksana@itl.rtu.lv*

Last data update December2013

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.ptsk.man.bialystok.pl*

leon@ibib.waw.pl

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

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

Last data update December2013

SIMS – Scandinavian Simulation Society

SIMS is the *Scandinavian Simulation Society* with members from the four Nordic countries Denmark, Finland, 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), DKSIM (Dansk Simuleringsforening) and NFA (Norsk Forening for Automatisering).

→ *www.scansims.org*

esko.juuso@oulu.fi

SIMS / Esko Juuso, Department of Process and Environmental Engineering, 90014 Univ.Oulu, Finland

SIMS Officers	
President	Esko Juuso, *esko.juuso@oulu.fi*
Vice president	Erik Dahlquist, *erik.dahlquist@mdh.se*
Treasurer	Vadim Engelson, *vadim.engelson@mathcore.com*
Repr. EUROSIM	Esko Juuso, *esko.juuso@oulu.fi*
Edit. Board SNE	Esko Juuso, *esko.juuso@oulu.fi*
Web EUROSIM	Vadim Engelson, *vadim.engelson@mathcore.com*

Last data update December2013

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 69 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 / Rihard Karba, 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	Aleš Belič, *ales.belic@sandoz.com*
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	Rihard Karba, *rihard.karba@fe.uni-lj.si*
Web EUROSIM	Vito Logar, *vito.logar@fe.uni-lj.si*

Last data update December2013

UKSIM - United Kingdom Simulation Society

UKSIM has more than 100 members throughout the UK from universities and industry. It is active in all areas of simulation and it holds a biennial conference as well as regular meetings and workshops.

→ *www.uksim.org.uk*

david.al-dabass@ntu.ac.uk

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

UKSIM Officers	
President	David Al-Dabass, *david.al-dabass@ntu.ac.uk*
Vice president	A. Orsoni, *A.Orsoni@kingston.ac.uk*
Secretary	Richard Cant, *richard.cant@ntu.ac.uk*
Treasurer	A. Orsoni, *A.Orsoni@kingston.ac.uk*
Membership chair	K. Al-Begain, *kbegain@glam.ac.uk*
Univ. liaison chair	R. Cheng, *rchc@maths.soton.ac.uk*
Repr. EUROSIM	Richard Zobel, *r.zobel@ntlworld.com*
Deputy	K. Al-Begain, *kbegain@glam.ac.uk*
Edit. Board SNE	Richard Zobel, *r.zobel@ntlworld.com*

Last data update December2013

EUROSIM OBSERVER MEMBERS

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.

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, an 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 December2013

ROMSIM – Romanian Modelling and Simulation Society

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

→ *www.ici.ro/romsim/*

sflorin@ici.ro

✉ ROMSIM / Florin Stanciulescu,
National Institute for Research in Informatics, Averescu Av. 8 – 10, 71316 Bucharest, Romania

ROMSIM Officers	
President	Florin Stanciulescu, *sflorin@ici.ro*
Vice president	Florin Hartescu, *flory@ici.ro* Marius Radulescu, *mradulescu@ici.ro*
Repr. EUROSIM	Florin Stanciulescu, *sflorin@ici.ro*
Deputy	Marius Radulescu, *mradulescu@ici.ro*
Edit. Board SNE	Florin Stanciulescu, *sflorin@ici.ro*
Web EUROSIM	Zoe Radulescu, *radulescu@ici.ro*

Last data update December2012

RNSS – Russian Simulation Society

NSS - The Russian National Sim ulation 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**.

→ *www.simulation.su*

yusupov@iias.spb.su

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

RNSS 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*
Deputy	B. Sokolov, *sokol@iias.spb.su*
Edit. Board SNE	Y. Senichenkov, *sneyb@dcn.infos.ru*

Last data update February 2012

SNE – Simulation Notes Europe

Simulation Notes Europe publishes peer revie wed *Technical Notes*, *Short Notes* and *Overview Notes* on developments and trends in modelling and simulation in various areas and in application and theory. Furthermore **SNE** docum ents the **ARGESIM Benchmarks** on *Modelling Approaches and Simulation Implementations* with publication of definitions, solutions and discussions (*Benchmark Notes*). Special *Educational Notes* present the use of modelling and simulation in and for education and for e-learning.

SNE is the official membership journal of **EUROSIM**, the Fede ration of Europe an Si mulation Societies. A News Section in **SNE** provides inf ormation f or **EUROSIM** Simulation Societies and Sim ulation Groups. In 2013, SNE int roduced an extended subm ission strategy i) individual s ubmissions of scientific papers, and ii) submissions of selected c ontributions from confere nces of EUROSIM societies for post-conference publication (suggested by conference organizer and a uthors) – both with peer review.

SNE is published in a printed version (Print ISSN 2305-9974) a nd in a n online ve rsion (Online IS SN 2306-0271). W ith **Online SNE** the publisher **ARGESIM** follows the **Open Access** strategy, allowing download of published contributions for free. Since 2012 **Online SNE** contributions are identified by an DOI (Digital Obj ect Identifier) assigned to the publisher **ARGESIM** (DOI prefix 10 .11128). **Print SNE**, high -resolution **Online SNE**, source codes of the *Benchmarks* and other additional sources are available for subscription via membership in a **EUROSIM** society.

Authors Information. Authors are i nvited to subm it contributions which have n ot been p ublished a nd ha ve not being considere d for pu blication else where t o the SNE Editorial Office. SNE di stinguishes different types of contributions (*Notes*):

- *Overview Note* – State-of-the-Art report in a specific area, up to 14 pages, only upon invitation
- *Technical Note* – scientific publication on specific topic in modelling and simulation, 6 – 8 (10) pages
- *Education Note* – modelling and simulation in / for education and e-learning; max. 6 pages
- *Short Note* – recent development on specific topic, max. 4 pages
- *Software Note* – specific implementation with scientific analysis, max 4 pages
- *Benchmark Note* – Solution to an ARGESIM Benchmark;basic solution 2 pages, extended and commented solution 4 pages, comparative solutions on invitation

Interested authors may find further information at **SNE's** website → **www.sne-journal.org** (layout templates for *Notes*, requirements for benchmark solutions, etc.).

SNE Editorial Office /ARGESIM

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

office@sne-journal.org (info, news)

eic@sne-journal.org Felix Breitenecker (publications)

EUROSIM 2016

9th EUROSIM Congress on Modelling and Simulation

City of Oulu, Finland, September 12 – 16, 2016

EUROSIM Congresses are the most important modelling and simulation events in Europe. For EUROSIM 2016, we are soliciting original submissions describing novel research and developments in the following (and related) areas of interest: Continuous, discrete (event) and hybrid modelling, simulation, identification and optimization approaches. Two basic contribution motivations are expected: M&S Methods and Technologies and M&S Applications. Contributions from both technical and non-technical areas are welcome.

Congress Topics The EUROSIM 2016 Congress will include invited talks, parallel, special and poster sessions, exhibition and versatile technical and social tours. The Congress topics of interest include, but are not limited to:

- Intelligent Systems and Applications
- Hybrid and Soft Computing
- Data & Semantic Mining
- Neural Networks, Fuzzy Systems & Evolutionary Computation
- Image, Speech & Signal Processing
- Systems Intelligence and Intelligence Systems
- Autonomous Systems
- Energy and Power Systems
- Mining and Metal Industry
- Forest Industry
- Buildings and Construction
- Communication Systems
- Circuits, Sensors and Devices
- Security Modelling and Simulation
- Bioinformatics, Medicine, Pharmacy and Bioengineering
- Water and Wastewater Treatment, Sludge Management and Biogas Production
- Condition monitoring, Mechatronics and maintenance
- Automotive applications
- e-Science and e-Systems
- Industry, Business, Management, Human Factors and Social Issues
- Virtual Reality, Visualization, Computer Art and Games
- Internet Modelling, Semantic Web and Ontologies
- Computational Finance & Economics
- Simulation Methodologies and Tools
- Parallel and Distributed Architectures and Systems
- Operations Research
- Discrete Event Systems
- Manufacturing and Workflows
- Adaptive Dynamic Programming and Reinforcement Learning
- Mobile/Ad hoc wireless networks, mobicast, sensor placement, target tracking
- Control of Intelligent Systems
- Robotics, Cybernetics, Control Engineering, & Manufacturing
- Transport, Logistics, Harbour, Shipping and Marine Simulation

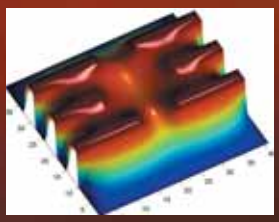
Congress Venue / Social Events The Congress will be held in the City of Oulu, Capital of Northern Scandinavia. The main venue and the exhibition site is the Oulu City Theatre in the city centre. Pre and Post Congress Tours include Arctic Circle, Santa Claus visits and hiking on the unique routes in Oulanka National Park.

Congress Team: The Congress is organised by SIMS - Scandinavian Simulation Society, FinSim - Finnish Simulation Forum, Finnish Society of Automation, and University of Oulu.
Esko Juuso EUROSIM President, Erik Dahlquist SIMS President, Kauko Leiviskä EUROSIM 2016 Chair

Info: eurosim2016.automaatioseura.fi, office@automaatioseura.fi

Parlez-vous MATLAB?

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



Modellierung eines elektrischen Potentials in einem Quantum Dot.

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

Image: Kim Young-Sang, Jung HeeJun, Quantum Device Lab, Hanyang Univ.

MATLAB®

The language of technical computing