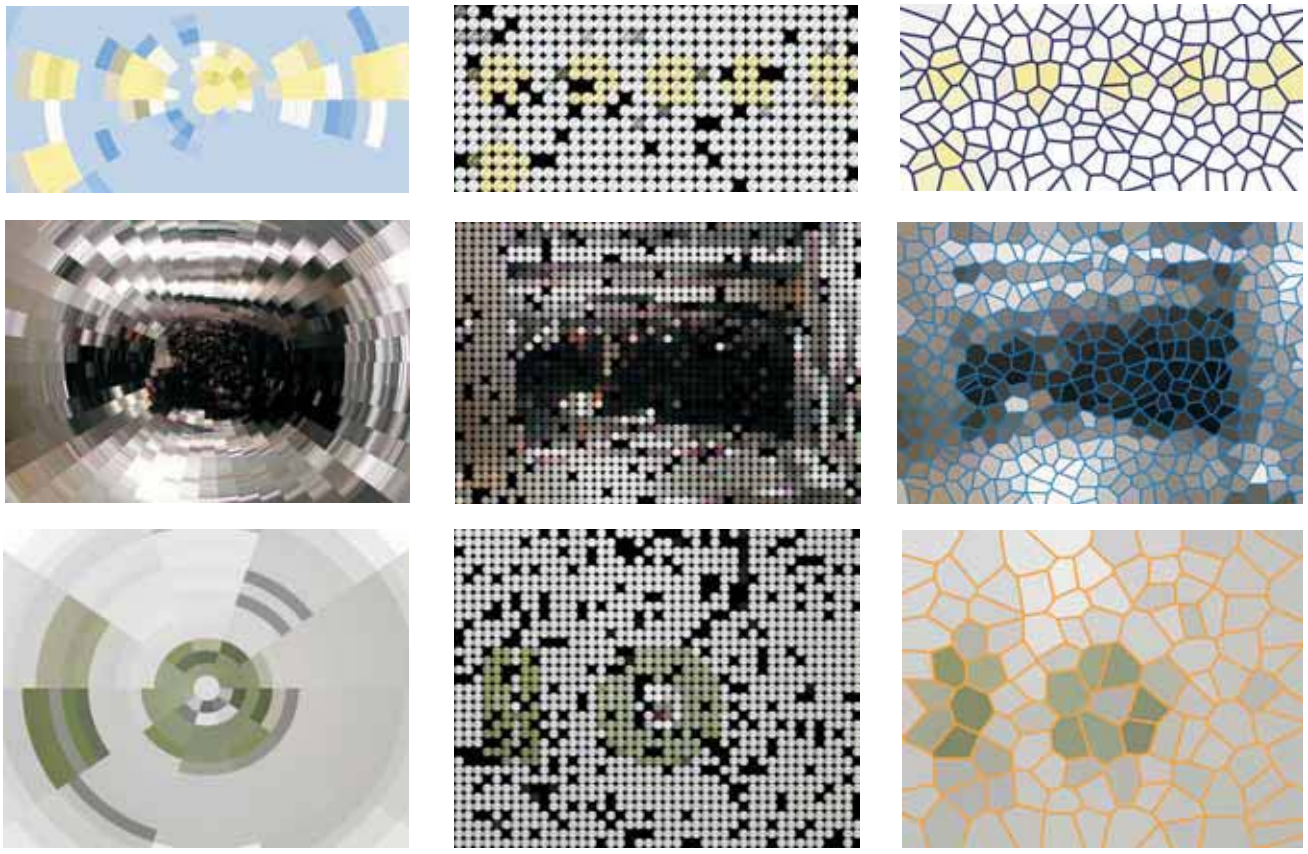
SNE SIMULATION NEWS EUROPE



Volume 16 Number 1 (SNE 46) August 2006, ISSN 0929-2268

Journal on Developments and Trends in Modelling and Simulation

Membership Journal for Simulation Societies in EUROSIM

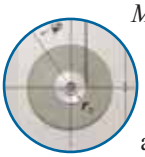
ARGESIM

Dear readers,
This is the second SNE issue with new layout, and we are glad, that we got positive reactions for changes in SNE layout and for opening the publication strategy of SNE. Together with this issue, we are proud to announce the first SNE Special Issue 'Parallel and Distributed Simulation Methods and Environments'. First born as idea in ASIM - ASIM Working Groups intend to publish alternately a Special Issue each year, the SNE Special Issues are open for all societies and conference organisers. The Special Issues cause a change in numbering the SNE issues: this regular SNE issue, SNE 46, is now identified as SNE 16/1 (Volume 16, Number 1), the first Special Issue as SNE 16/2; the next regular SNE double issue (SNE 47/48) will be numbered SNE 16/3-4. This remembers, that we are running SNE since 16 years, and we thank our faithful readers.
Together with the new layout, both editorial boards are being reorganised and will be enlarged for the future. We are also working on a new infrastructure for running an editorial office, together with tasks for SNE on the web.
We hope, the readers enjoy this issue, and the contributors appreciate the new editorial structure (more strict, but hopefully more efficient). Three Technical Notes and three Short Notes in this issue show the broad variety of modelling and simulation. The Technical Notes are special ones: based on a post-conference review procedure via Internet for contributions to MATHMOD 2006 Vienna, papers were selected for publication in SNE (to appear also in the next SNE issues). Furthermore, as first reaction on the ARGESIM / MATHMOD Yo-yo Challenge, the Techncial Note by Leon Zlajpah introduces into mechanical mysteries of Yo-yo control. The Comparison Section publishes an updated version of Comparison C13 'Crane and Embedded Control', reflecting the developments in this area of modelling and simulation; furthermore, seven comparison solutions concentrate on modelling issues and alternative approaches.
The News Section reports about progress in new structures for EUROSIM, and about activities in EUROSIM member societies and in Societes related to Modelling and Simulation. We thank all contributors, members of the editorial boards, and people of our ARGESIM staff for co-operation in producing this SNE issue.

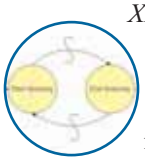
Felix Breitenecker, editor-in-chief; Felix.Breitenecker@tuwien.ac.at

SNE 16-1 / SNE 46 in Five Minutes

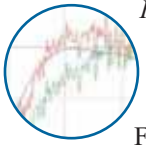
Process Modelling in a Sterilisation Tunnel (TN) - presents modelling and simulation for temperature profiles in an industrial production process – ***page 3***



Modelling and Control of Yoyo (TN) – deals with the classical Yoyo toy: mathmatical models for control and for haptic interfaces, control strategies, and verification by a robot – ***page 9***

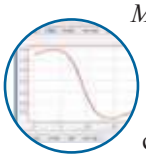


XML in DEVS (TN) – introduces XML as model basis for discrete event models for simulation via WWW and presents a prototype implementation – ***page 16***

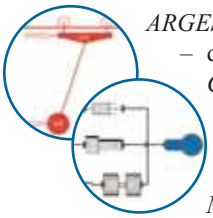


Real-time Simulation with DSPs (SN) - reports about a connection of two DSPs, one identifying the plant, the other performing Kalman Filter and LQ control – ***page 21***

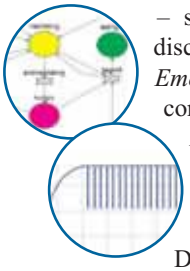
Simulation of Blood Glucose Regulation (SN) – presents MATLAB models glucose status together with a graphical interface for educational use – ***page 23***



Modelling and Control of a 2DOF - Robot (SN) – outlines modelling and simulation of a simple robot for E-learning of simulation and control via WWW – ***page 25***



ARGESIM Comparison Section – defines a revised benchmark C13 *Crane and Embedded Control* (implicit modelling, digital control, sensor action), followed by a sample solution with *Modelica/Dymola* – ***page 27***



– seven *Comparison Solutions* for discrete comparisons (*Dining Philosophers*, *Emergency Department*), continuous comparisons (*Switching States*, *SCARA Robot*) and general comparisons (*Cellular Automata*, *Identification*) show efficient implementations using MATLAB/Simulink, Dymola, DSOL/Java, Maxima and special Petri Net tools – ***page 31 - 38***

Book Reviews and *Journal News* – Eleven book reviews and one book news

Introduction of the SNE Special Issue *Parallel and Distributed Simulation Methods and Environments*

Call for next SNE Special Issue *Validation and Verification*
– ***page 39 - 47***

Young Simulationists – introduction of simulationists from Germany and Austria – ***page 48***

EUROSIM Society Reports – 20 pages reports from EUROSIM societies, followed by 8 pages from International Societies and Groups (ECMS/SCS, MATHMOD, Modelica, etc.) and 2 pages *Industry News* in the *News Section*

Content

SNE Editorial Boards

SNE - Simulation News Europe – is advised by two Editorial Boards. The *SNE Editorial Board* is taking care on reviewing and handling of Technical Notes, Shortnotes, Software Notes, Book and Journal Review, and of Comparison and Benchmark Notes. The *SNE News Editorial Board* (News Section) is responsible for reports from EUROSIM, EUROSIM societies, International Societies, and for Industry News.

Editorial board

Contact

SNE - Editors /ARGESIM
c/o Inst. f. Analysis and Scientific Computation
Vienna University of Technology
Wiedner Hauptstrasse 8-10, 1040 Vienna, AUSTRIA
Tel + 43 - 1- 58801-10115 or 11455, Fax - 42098
sne@argesim.org; WWW.ARGESIM.ORG

Editorial Info - Impressum

SNE Simulation News Europe ISSN 1015-8685 (0929-2268).
Scope: Information on development in modelling and simulation, comparisons and bechmarks on modelling and simulation, membership information for EUROSIM and Simulation Societies.
Editor-in-Chief: Felix Breitenecker, Mathematical Modelling and Simulation, Inst. f. Analysis and Scientific Computing, Vienna University of Technology, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria; *Felix.Breitenecker@tuwien.ac.at*
Layout: A. Breitenecker, ARGESIM TU Vienna / Linz; Anna.Breitenecker@liwest.at
Printed by: Grafisches Zentrum, TU Wien, Wiedner Hauptstrasse 8-10, A-1040, Wien
Publisher: ARGESIM and ASIM
ARGESIM, c/o Inst. for Scientific Computation, TU Vienna, Wiedner Hauptstrasse 8-10, A-1040 Vienna, Austria, and ASIM (German Simulation Society), c/o Wohlfartstr. 21b, D-80939München

A Programmed Solution to ARGESIM Comparison C 6 'Emergency Department' with DSOL, a Java- based Suite

Roland Lezuo, Felix Breitenecker, Vienna Univ. of Technology, Austria; *fbreiten@osiris.tuwien.ac.at*

Simulator: *DSOL* is an open source, Java based suite for continuous and discrete event simulation. *DSOL* is written with distributed computing in mind and supports distributed models. Models are implemented in Java code. *DSOL* offers facilities for graphical data evaluation like charts but also supports 2D and 3D animation. There are predefined classes for standard entities like generators, stations and resources. There is also a big library of statistic classes, including distributions and tallies which compute mean and standard deviation automatically.

Model / Implementation of Strategies for tasks. Model and experiments (tasks) of C6 are implemented using one class for each station (*registration*, *casualty ward*, *x-ray* and *plaster*). There is just one casualty ward object hiding the details of two wards with two doctors each. Additionally a patient source and a patient sink are used, primary for evaluation purposes and implementation details. The patients themselves are a class also and know the details about their path through the hospital and their priority for task c. The glue between all the classes is DSOL's event mechanism.

The patient source schedules a `goto next station` event for newly created patients and re-schedules the `create patient event` as long as needed.
The patient schedules an arrival event on his next station and gets queued there. As soon as the station treated the patient it will create an `goto next station` event for the patient and reschedule itself as long as there are patients queueing. The patient sink terminates this mechanism and stops the simulation when all patients left the hospital.

The method `getNextStation` returns the next target of the patient. In that way only the patient has to know about his way through the hospital, so new types of patient could be added easily.

Each station implements a `queuePatient` method, basically re-scheduling itself: in pseudo code:

```
void queuePatient(Patient p)      {
  if (queue.isEmpty())  {
   queue.add(p);
    scheduleProcessing(drawStationDelay());
 } else{
    queue.add(p);    } }
```

The last missing puzzle piece is the implementation of `scheduleProcessing` which draws a random delay and after that delay calls a method named `processing`. This method in pseudo code:

```
void processing()                 {
  p = queue.pop();
  p.notify();
  if (!queue.isEmpty())       {
   scheduleProcesing(drawStationDelay());}}
```

It basically takes the first patient from the queue and puts him to the next station. If there are still patients left it is re-schedules itself. The patient sink finally stops the mechanism by implementing a method, which writes statistically data to the experiment database (to be examined by the *DSOL GUI*).

Tasks a -c: Different strategies for operation. As special actions (change of doctors) and priorities are programmed directly in the method definitions, no further effort is necessary for complete the more complex tasks b and c.

Results for 50 simulation runs are given in Table 1 below. In general, results from task b are almost identically to task a, so swapping doctors seems to have no effect. Task c shows improvements of the priority strategy: although the overall treatment time for all patients did not change, the overall treatment time for each patient is reduced.

Type	sm[a]	sd[a]	sm[b]	sd[b]	sm[c]	sd[c]
1	205	12.47	215	13.16	169	14.97
2	98	14.13	99	12.63	109	15.68
3	286	22.22	287	17.52	165	21.63
4	98	4.81	106	5.97	122	6.18
all	171	13.41	176	12.32	141	14.62

Table 1: Mean and standard deviation for treatment times.

Classification: Programmed DEVS library-based approach.

Corresponding Author: Felix Breitenecker
Roland Lezuo, Felix Breitenecker, Vienna Univ. of Technology, Inst. f. Analysis and Scientific Computation, Wiedner Hauptstrasse 8-10, 1040 Vienna, Austria;
fbreiten@osiris.tuwien.ac.at

Received: October 24, 2005
Revised: January 23, 2006
Accepted: March 30, 2006