THE ART OF DATA-DRIVEN MODELLING IN LOGISTICS SIMULATION

Rainer Frick

V-Research GmbH, Dornbirn, Austria

rainer.frick@v-research.at

ABSTRACT

The field of discrete event simulation used for logistics simulation is covered by many software tools and software libraries. Most of them offer libraries and functionality to model a simulation case and run experiments. Already small changes of the logistics case can result in considerable efforts of changing the simulation model. Data-driven simulation modelling provides the possibility to reduce adaptation work and makes simulation model more long-lasting. This work comes up with a reflection of state-of-the-art data-driven modelling techniques and reports on implemented sample cases in the world of logistics.

Keywords: Simulation, Modelling, Data-driven

1. INTRODUCTION

Discrete event simulation is part of operations research and is concerned with modelling dynamic systems. The system is described by time-dependent state variables and these variables are changed by occurring events. Simulation helps modelling more complex process correlations and can be used for comparative analysis (März 2008, Wenzel 2009).

In the logistics field simulation is used to model process flows to analyze cost structures, to discover bottlenecks or to visualize the business activities behind the scene, i.e. of transportation networks, warehouse activities or production flows.

To create a valid and significant simulation model in these areas can be a cost-intensive matter. Simulation know-how and knowledge in the specific area must be combined to shape an appropriate simulation entity and process model. And what happens if the underlying business case changes or enhances? This requires adaptations in the simulation model as well and therefore a simulation expert is needed again which increases costs.

Data-driven modelling of simulation studies provides more flexibility in terms of a changing environment.

Partly data-driven modelling can be found in many simulation studies. Normally order, product or demand data is taken out of a data archive for their usage in a simulation. Layout and resource data in a simulation model are represented as simulation entities or objects which are most often created by drag and drop in a graphical editor.

2. THE APPROACH

Presently modelling of a simulation study is done by defining simulation entities in a simulation software tool with the help of a graphical editor (Figure 1) or implementing them directly in code with the specific programming language.

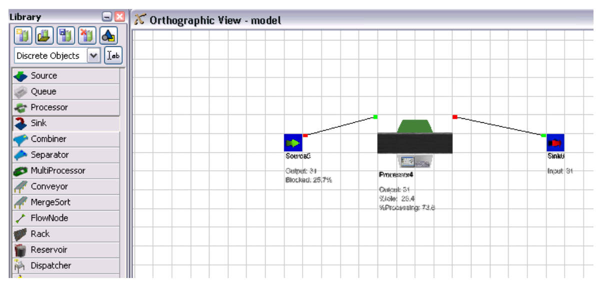

Figure 1: Model-driven modelling

As a sample can serve a warehouse where racks, forklifts and workers are placed in the simulation model and interconnected with each other (see Figure 2).

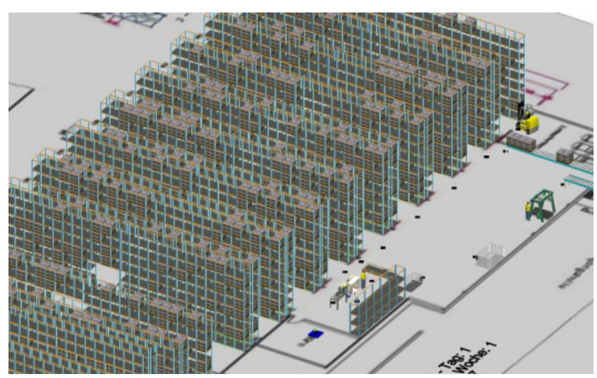

Figure 2: A modelled sample warehouse

Later changes of the model in Figure 2, i.e. placing more racks, mean changing the simulation model itself in the simulation software tool. This requires a simulation expert again, because the additional racks have to be included into the warehouse layout and workflow.

A dynamic way of building the simulation model automatically can obviate this change loop. New simulation entities are added to a database and not directly to the simulation model. Therefore the basic concept of data-driven modelling is the separation of the simulation model from the used simulation software. All the data needed for a simulation study are lodged in a data archive, most often a database, and not in the simulation software. At runtime the data is read from the archive and the simulation model is created dynamically. In other words the simulation entities necessary to run a simulation with a specific simulation software tool are created with the help of a generation software tool out from the data archive (Wang 2008, Hassan 2009, Jensen 2007).

Figure 3 shows the software components needed most often for data-driven modelling of a simulation study. All the data necessary for the simulation model is stored in a data archive. For storing, changing and viewing this software an assistance software tool can be helpful. The generation software creates the simulation model able to run out from the archive. This model is carried out by a simulation software tool (Yang 2008).

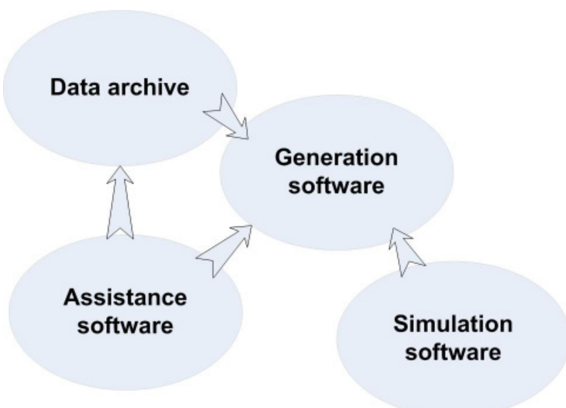

Figure 3: Components of a data-driven modelling system

As a data archive often a relational database or Xml files are used. The generation software is an individually implemented tool focused on the specific problem domain with knowledge of the used simulation software. It provides the simulation entities for the simulation software. The simulation software can be any commercial or non-commercial simulation tool with an application programming interface (API) (Bergmann 2010).

Figure 4 demonstrates the basic application cycle of a data-driven simulation. Common for all simulation studies is the phase of data collection and preparation. A data model is defined, a data archive created and the data stored in tables, files etc. Out of this data the simulation model is generated automatically and it is executed by a simulation software tool. During simulation operation steps, tasks, events, times are logged into the data archive. With this logging information the results can be calculated and the result analysis is prepared. This result analysis leads to scenarios deduced from the original model. Alternative experiments can be carried out.

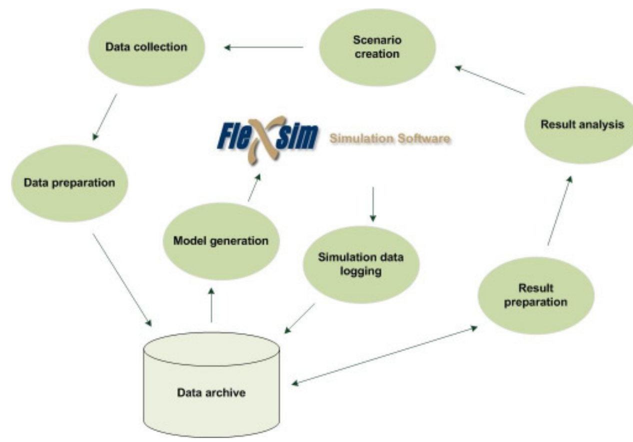

Figure 4: Simulation application cycle

3. PROCESS LEVEL

Data-driven modelling provides the simulation entities automatically to be used in the simulation model. But how about the process flows, the mode information and material find their way?

Three modes are possible.

First the process flow is not part of the data archive and the generation software. It is implemented in the simulation software and only the simulation entities are created automatically by the generation software.

Second the process flow is defined implicitly by creating the simulation objects. In other words the behaviour of these objects is implemented in their abstract definition, normally in a particular code class. This is common practice in agent-based simulations.

Third the process flow is generated automatically too and therefore must be expanded into the data archive. To model the process flow the assistance software has to be extended to define the interaction between the simulation entities; a process model editor is suggested to handle this. Therefore process units must be defined, each of them combined with some simulation entities. The process model editor is capable to combine these process units to a valid and complete process flow (see Figure 5). To establish alternative process arms conditions which interpret the state of simulation entities are needed.

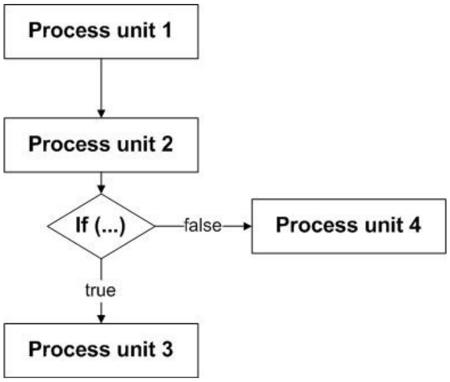

Figure 5: Sample process flow

4. PROS AND CONS

Data-driven modelling offers more flexibility to the process of building simulation models. Changes in the simulation case can be handled by the generation software to build a model which is able to run. In some circumstances the modelling data can be taken directly from the enterprise planning software which means an integration of the simulation technique into the operational business. Some other less important pros can be identified like the relieving of the computer system by loading only necessary data at runtime, the usage of software design patterns by separating programming code into different aspects, consistency check of the data archive and a step towards to a modelling standard.

But ever more flexibility is caught in the generation software its complexity raises to a higher level which can make their implementation more difficult. Nevertheless it is worth the effort because the costs have to be seen in relation to the practicability of the resulting simulation models.

5. LOGISTICS SAMPLES

In the scope of the authors daily work view simulation projects have been settled down with different data-driven principles behind them.

5.1. Transportation network

For a less complex simulation problem it can be helpful to do the generation of the model in the simulation software tool itself. Therefore only a database access has to be existent in the tool; no assistance software tool (see Figure 3) is needed. A transportation network in Slovakia serves as a sample (Figure 6). The branches, transport resources and other necessary structural data is kept in a database, extracted by the generation part of the simulation software tool and brought to a simulation model. The process flow is hard-coded in the simulation tool itself in a special simulation library. The results of the simulation can be interpreted in key figures such as utilization of resources, shipment times and more (Levinson 2004).

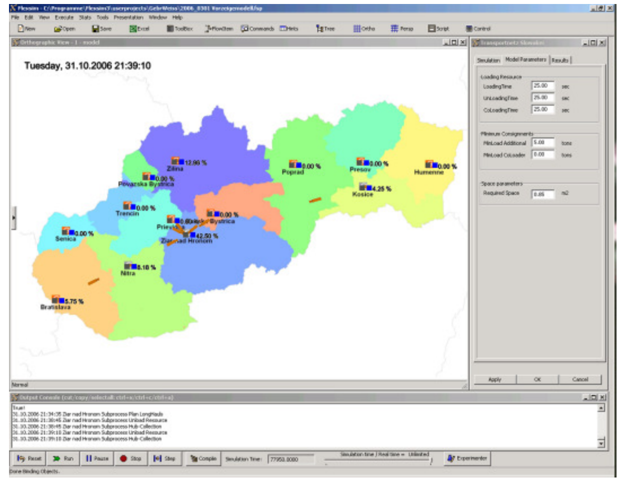

Figure 6: Transportation network sample

The data model is kept easily, the generation complexity straightforward and the process flow not very adaptable but a wide range of transportation networks with its long- and short-distance subtleties can be modelled and analyzed rapidly.

A further extension of this sample is to bring the process flow into the simulation objects it selves. This leads to more manageable and adaptable software code.

5.2. Warehouse

With increasing complexity of the logistics case the data-driven generation of a simulation model becomes more complex as well. This can be seen in a warehouse simulation study (Figure 7). An assistance software tool is essential here and it can be seen that the generation part has to be separated from the simulation tool. This is a crucial advantage for further adaptations to the logistics situation.

Each the warehouse layout and the process flow are modelled outside of the simulation tool by a layout and a process model editor. The generation software combines the layout and the process flow to an executable simulation model.

The generation software in this sample inherits even more functionality by providing schedules for the assignment of warehouse resources and workers. This is useful if own implementations of schedules and strategies shall be applied.

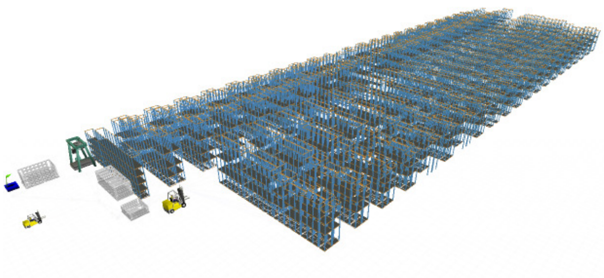

Figure 7: Warehouse sample

5.3. Production plant

A simulation software tool can be only a small part of the overall analysing and visualizing system. As seen in Figure 8 which demonstrates a production line. The modelling information about plant units and machines comes from a layout data archive and is translated into simulation objects. The tasks for the production line come from a production server. The simulation tool is responsible for the 3D visualization, the scheduling of the production tasks and the logging of operation data and time. The generation part is quite complex but very adaptable. New production lines can be modelled really quickly. Plant cycle times and machine cycle times can be analyzed via the logged data.

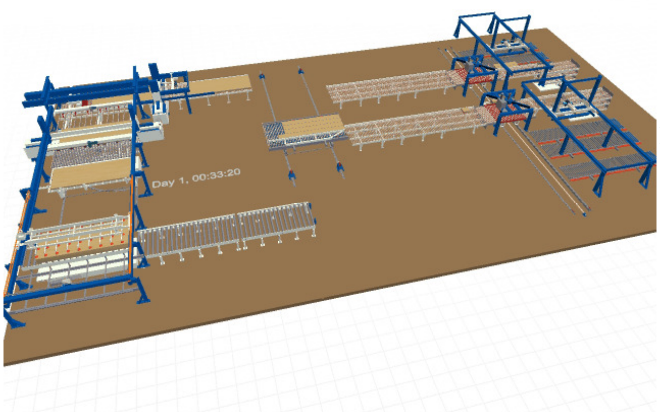

Figure 8: Production plant sample

6. CONCLUSION

The simulation of logistics cases helps to achieve a better understanding of the processes taking place. Key figures can be evaluated and different action alternatives can be compared to get a better performance in some respects.

A software framework and data-driven modelling reduce simulation model building time tremendously (Jensen 2007). An additional process editor enhances this technique and provides the capability of separating the structural model data from the operational flow and underlines the possibility of modelling distinct versions of process flow alternatives for a simulation study to compare them regarding some predefined key figures.

Taking altogether data-driven modelling can be a crucial advantage in changing logistics environment.

REFERENCES

David Levinson. The evolution of transport networks. Technical report, 2004.

Sören Bergmann, Alexander Fiedler, and Steffen Straßburger. Generierung und Integration von Simulationsmodellen unter Verwendung des Core Manufacturing Simulation Data (cmsd) Information Model. *Integrationsaspekte der Simulation: Technik, Organisation und Personal*, 2010.

Samer Hassan, Luis Antunes, and Juan Pavon. A data-driven simulation of social values evolution. *8th International Conference on Autonomous Agents and Multiagent Systems*, 2009.

Sigrid Wenzel. Modellbildung und Simulation in Produktion und Logistik, Stand und Perspektiven. *ASIM Workshop*, 2009.

Minghui Yang. Using data driven simulation to build inventory model. *Proceedings of the 2008 Winter Simulation Conference*, 2008.

Sven Jensen. Eine Methodik zur teilautomatisierten Generierung von Simulationsmodellen aus Produktionsdatensystemen am Beispiel einer Job Shop Fertigung. *ASIM Workshop*, 2007.

Catriona Kennedy, Georgios Theodoropoulos: Towards intelligent data-driven simulation for policy decision support in the social sciences. *School of Computer Sciences, University of Birmingham, UK.*

Lothar März, Martin Saler: Ein Analyse-, Planungs- und Entscheidungsinstrument für Lagerlogistikanwendungen, *Advances in simulation for production and logistics applications, Markus Rabe (Hrsg.), Stuttgart 2008.*

Sigrid Wenzel, Anke Jauss: Diskussion eines Benchmark-Verfahrens für den Vergleich von Simulationswerkzeugen in Produktion und Logistik, *Advances in simulation for production and logistics applications, Markus Rabe (Hrsg.), Stuttgart 2008.*

Guixiu Qiao, Frank Riddick, and Charles McLean. 2003. New manufacturing modeling methodology: data driven design and simulation system based on XML. In *Proceedings of the 35th conference on Winter simulation: driving innovation* (WSC '03). Winter Simulation Conference 1143-1148.

AUTHORS BIOGRAPHY

Rainer Frick has studied Computer Science and is currently studying Business Administration at the University of Innsbruck. His main focus is developing software in the simulation and optimization field at V-Research company.