Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia CIRP 7 (2013) 455 – 460

www.elsevier.com/locate/procedia

Forty Sixth CIRP Conference on Manufacturing Systems 2013

Methodology and data-structure for a uniform system's specification in simulation projects

Csaba Kardos[a]*, Gergely Popovics[a,b], Botond Kádár[a], László Monostori[a,b]

[a]Fraunhofer Project Center for Production Management and Informatics at The Computer and Automation Research Institute, Hungarian Academy of Sciences, Kende u. 13-17, Budapest 1111, Hungary

[b]Budapest University of Technology and Economics, Műegyetem rkp. 1, Budapest 1111, Hungary

* Corresponding author. Tel.: +36-1-279-6181; fax: +36 1 4667 503.*E-mail address*: csaba.kardos@sztaki.mta.hu.

Abstract

In the last few decades the evaluation and analysis of manufacturing systems' behavior and their performance became very important. Digital enterprise technologies, as for example discrete-event simulation (DES), are effective tools both in production related decision making processes and in structure and performance analysis of manufacturing systems. However, building a discrete-event based simulation model of a manufacturing system is a difficult task and requires special competence. The majority of simulation studies are aimed at analyzing a certain problem by a specific simulation model created by experts with a relatively high financial expenditure. The paper introduces an ongoing research aimed at developing a framework to reduce the efforts spent on draft simulation studies by simplifying and accelerating the process of model building. The proposed modeling methodology uses a uniform data structure which is a production oriented implementation of the ANSI/ISA-95 standard and supports the creation of models without simulation software specific knowledge. The supporting data structure enables the development and application of proprietary simulation engines tailored for specific problems. The paper compares the traditional and the proposed methodologies and also introduces the first experiments gained on specific test-cases. In our approach the simulation models are created automatically and independently from simulation tools which will be presented through the examples of both commercial and self-developed applications.

Selection and peer-review under responsibility of Professor Pedro Filipe do Carmo Cunha

Keywords: production simulation; ISA 95; selfbuilding simulation; data structure;

1. Introduction

In the last few decades Digital Enterprise Technologies (DET) have become powerful tools in analyzing the behavior and evaluating the performance of manufacturing systems. As a part of DET Discrete-Event Simulation DES also has its share in supporting analysis and decision making regarding manufacturing systems and processes. [6, 8]

Advantages and benefits provided by the application of DES are described in numerous scientific publications (i.e. [2, 6, 8, 9]) and they can be summarized as follows:

- Better overview and understanding of the system and processes during the model building phase.
- Increased throughput, decreased process times, reduced in-process inventories of parts, increased utilizations of resources, reduced capital requirements or operating expenses.
- „Virtual" statistical data by analyzing results from simulation.

Unfortunately, realizing these benefits and running successful simulation projects are often restricted by different factors. These factors are epitomized in [4] and they are labeled as "grand challenges". The description of these grand challenges in accordance with different papers [10, 11] emphasizes the resource consuming nature of simulation projects in terms of time and expertise, which at the end results in costly simulation projects.

Since the lack of funds is a crucial point for Small or Medium Enterprises (SMEs), running simulation projects might be unaffordable for them. Although in many cases, considering the complexity of the modeled

2212-8271 © 2013 The Authors. Published by Elsevier B.V.
Selection and peer-review under responsibility of Professor Pedro Filipe do Carmo Cunha
doi:10.1016/j.procir.2013.06.015

processes and manufacturing systems, application of simulation would be desirable for SMEs as well.

To underline the importance of SMEs in the economy, it has to be noticed that in the countries of the Organization for Economic Co-operation and Development (OECD), the number of the SMEs are 99,9% of the total number of enterprises [3]. In Hungary, 45% of the GDP is given by SMEs, and more than 60% of all employees work in this sector [3]. Even if their manufacturing capabilities and process complexity cannot be compared to larger enterprises, based on these facts it is reasonable to provide tools, which efficiently support simulation projects for SMEs.

The goal of the research to be outlined here is the implementation of an environment which offers quick and easy-to-use solution to perform rough analysis and evaluation of the behavior and performance of manufacturing systems by applying DES. The subject of the paper is to introduce our efforts creating a system, called EasySim, which is aimed at accomplishing this goal.

2. Implementing simulation projects

To fulfill the defined goals the resource consumption of the simulation project has to be reduced. In order to achieve this, the detailed inspection of simulation projects is required.

Figure 1 shows the usual distribution of efforts among the tasks of a simulation project. It can be concluded that data acquisition requires almost the quarter of the whole project timespan, while modeling requires even more. As these two activities consume the majority of time and resources, reducing the resource-consumption of these phases can have significant impact on the efficiency of the whole project [4].

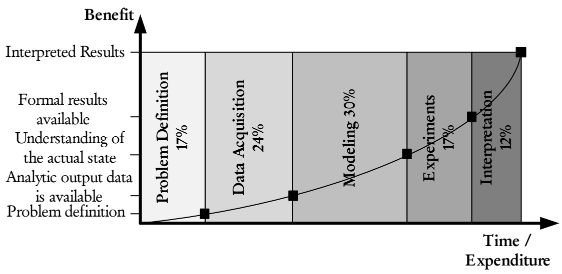

Fig. 1: Usual work-content distribution and the development of results and benefits in simulation projects [2].

Figure 2 shows the beginning stages of conventional simulation projects (since the paper does not focus on the results analysis, these stages are not included in the drawing.).

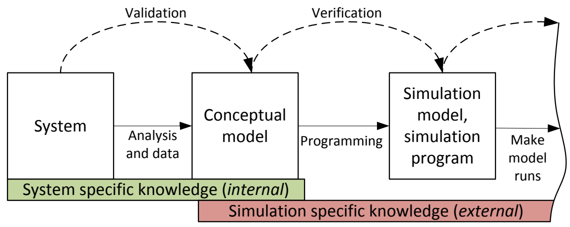

Fig. 2: Beginning stages of conventional simulation projects (based on [8] and [9])

The figure also depicts the usage of another main resource, which is highlighted in this paper, namely the application of expert knowledge. In the beginning of simulation projects, the real system (or its specification) is given, which has to be mapped to a conceptual model through detailed data acquisition and analysis and by applying simplifying assumptions [8, 9]. Building up a valid and reliable conceptual model is a crucial point, which the success of the whole simulation project relies on. However, carrying out this activity requires both system specific knowledge and simulation specific knowledge. Unfortunately, it is rarely the case that an internal expert of the system is identical to the person who is capable to build a proper simulation model. Therefore, from the conceptual model building phase, simulation experts are also involved into the project. On one hand this is costly, while on the other hand requires successful cooperation between different experts. An example of such cooperation is the definition of the proper level of model details. This highly influences the credibility of the simulation results (system aspect) and in parallel it also affects the speed, as much, the usability of the simulation model (simulation aspect), thus a balanced solution has to be found. [8, 9].

Based on the conceptual model, the task of simulation experts is to translate it into a simulation model. This can be carried out either by using a specific simulation software product or language (e.g. Arena, Plant Simulation or Simula) or by using a general purpose programming language (e.g. C#, Java). As any programming activity, this is often an error-prone and time consuming task. In order to avoid the above discussed drawbacks a possible solution is the application of automatic simulation model building technologies.

Papers regarding this topic describe two different procedures. The first is tailored to gather input data automatically from the data-sources available on different levels of manufacturing systems (i.e. Manufacturing Execution Systems (MES), Programmable Logic Controllers (PLC) programs) and builds the simulation model automatically based on these data [11]. Nevertheless, the implementation of automated data gathering is also a very time-consuming task and the availability (and accessibility) of these data-sources can be very limited, especially in case of SMEs.

The second approach defines guidelines for the conceptual modeling, so that it could be used as a source for automatic model building [10]. The application of this approach is more suitable, when there are limited data sources or the size of the model makes application of the first approach inefficient.

3. Components of automatic simulation model building

By the application of the previously described idea of automatic simulation model building, two main advantages can be achieved compared to the manual model building:

- The process of simulation model building can be simplified and speeded up.
- The verification of the simulation model is not necessary, since it is built up without any user interaction within the automatic model building environment.

Comparing Figure 2 to the Figure 3 it can be seen that the process of automatic model building comprises both the conceptual model and the simulation model. The interpretation of the conceptual model is carried out by predefined algorithms. As the result of this concept the simulation project can be carried out with significantly less simulation specific knowledge, since these tasks are enclosed in a predefined environment.

Nevertheless these advantages come at a price: the algorithms, which handle the input data and build up the simulation model, have to be fed with properly structured data. According to this an essential part of the proposed automatic model building environment is the definition and application of a uniform data structure.

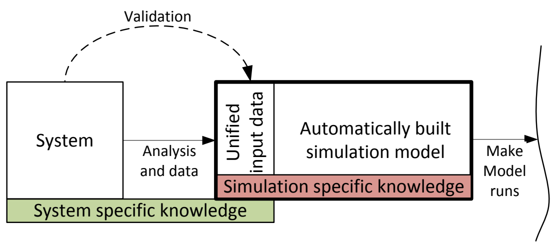

Fig. 3: The beginning stages of a simulation project, which is supported by automatic model building.

Since creating simulation models requires the integration of information coming from different levels, the uniform data structure has to provide a standard set of elements, which can represent all the relevant components of manufacturing systems and their relationships. According to its definition: "ISA-95 is an international standard for the integration of enterprise and control systems" [5]. Based on this definition and taking the fact that the standards introduced by ISA (International Society of Automation) are widespread and commonly used, it is a reasonable choice to apply ISA-95 for defining the uniform data structure [7].

The standard supplies models and terminology that define the information to be exchanged between the planning and execution levels of a production system [5]. Figure 4 shows the different levels of manufacturing systems (e.g. production plans, dispatching rules, operation and machine specific details) and how uniform data structure connects them.

The goal of the research is not only to create uniform conceptual representation of manufacturing systems, but also to use this uniform representation binding together the conceptual and the simulation model.

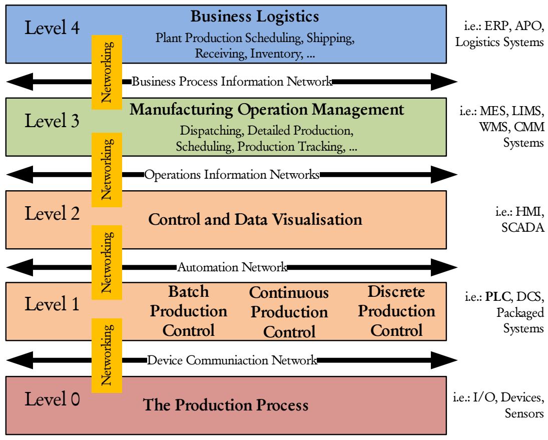

Fig. 4: The different levels of an industrial enterprise as described in the ISA-95 standard. The method introduced in this paper requires the integration of data from level 2, 3 and 4.

3.2. Structure of the proposed simulation model

Following the guidelines of the standard the proposed representation is capable to describe a Job-shop manufacturing system, which includes the following components.

There are three basic categories of elements: equipment, process segments and product segments. The detailed definition and the relationships of these elements can be given by using *classes*, class *properties*, *dependencies* and element *specifications*, as shown in Figure 5.

- *Classes* define instances of elements, e.g. a buffer class defines instances of equipment elements, which store product without processing them.
- *Process segments* represent time dependent processes, which can be carried out on equipment elements.

- *Specifications* link different elements, e.g. equipment element can be linked to process segment by defining *EquipmentSegmentSpecification.*
- *Dependencies* can describe series of consecutive segments. For example a production routing can be represented by a *ProcessSegmentDependency*, which is composed by a series consecutive process segments.

Using the above introduced elements the static structure of the manufacturing system can be characterized, however, the dynamic behavior of the simulated system should also be given.

The *type* attribute sets the behavior of equipment elements regarding the time overlapping of the executed processes:

- Single equipment can only handle one process at a time.
- Parallel Equipment can handle multiple processes at a time (according to its capacity), while additional attributes (subtypes) define the time overlap of the performed parallel processes, e.g. batch, parallel, line.

Moreover, the time-dependent attributes of the operations performed on equipment elements also have to be defined, e.g. mean time to repair, mean time to failure, availability, rejection rate, scrap rate. Nevertheless many of these attributes can be described redundantly, as for example the rejection rate can be both machine and process specific value. Therefore there is a hierarchy between the redundant attributes, which specifies the order of selection among them.

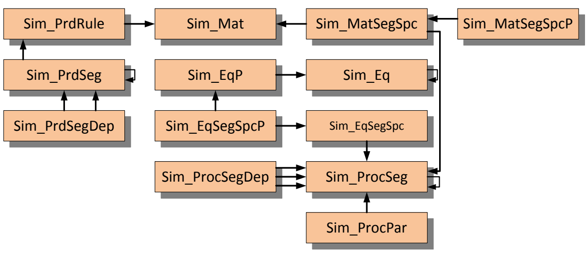

Fig. 5: A segment of the class diagram of the proposed ISA-95 based data structure.

In order to be able to evaluate and compare the performance of manufacturing systems uniform data structure also provides a set of Key Production Indicators (KPIs).

4. Introduction of the EasySim system

Applying the previously introduced uniform data structure allows the creation of a modular simulation environment. This, therefore, can be easily customized in order to fulfill the defined goals of the research. Figure 6 depicts the structure of the EasySim system and emphasizes this concept of modularity.

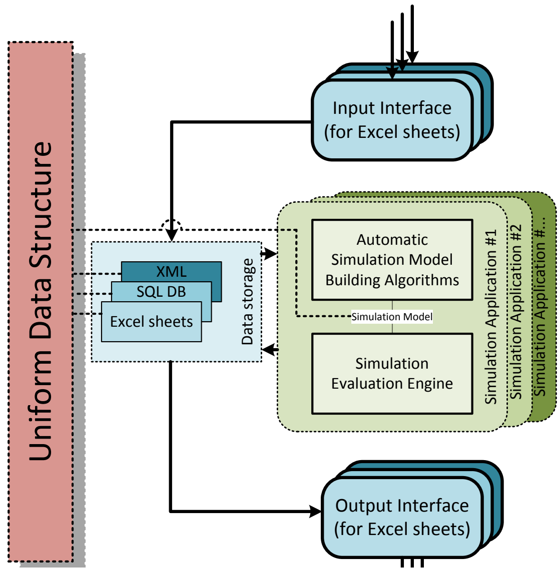

Fig. 6: The structure of the EasySim framework.

As it is highlighted in Figure 6 the system is comprised of four main components, which are the following:

- input interfaces,
- output interfaces,
- data storage,
- simulation application.

The guidelines of the uniform data structure bind these components together ensuring that the exchanged data is in the proper format.

The input handling interface provides environment for data preprocessing and it reads and forwards the data to the data storage component in the uniform data structure. Whereas staying simple is one of the baselines of our development, primarily Microsoft Excel was used as input interface. It is a very prevalent tool (even in very low profile enterprises), its usage does not require advanced skills and in addition it offers sufficient capabilities as well as to handle and to store the defined tables of the uniform data structure. However, in cases of more complex manufacturing systems, where the representation of the system can be large sized datasets, a more enhanced data handling solution might be required. In order to provide solution for these cases, the uniform data structure is also implemented by using a defined SQL database and XML structure. These are more powerful tools, which perform better in handling and storing large datasets, though — beside the definition of the structure — they require the implementation of a standalone user interface.

The simulation application contains the algorithms, which implement the procedure of automatic model building from the stored uniform data. The simulation model is evaluated by DES algorithms, which calculates the defined KPIs of the system and returns the results to the uniform data structure.

The output interface reads the calculated results from the data storage component and visualizes them.

4.1. Self-developed simulation engine

The uniform data structure not only supports the usage of different input and output interfaces, but it also allows operating multiple simulation engines. As long as both the automatic model building algorithms (in order to handle the uniform input data) and both the calculation of the defined KPIs (in order to support the uniform output data structure) are implemented, the simulation applications are replaceable.

In our research two different simulation applications were applied. Siemens Plant Simulation is a commercial DES software tool, which is a general purpose product with versatile modeling capabilities. It also offers its own programming environment to develop the automatic simulation building algorithms. It can be executed through parameterized batch files and therefore it can serve as a simulation engine. However, using a general purpose commercial product encloses the issues mentioned in the introduction. For SMEs with relatively simpler manufacturing systems, purchasing a commercial DES application and using only a relatively small part of its capabilities is not cost-efficient.

In order to create a more tailored application, development of a discrete event simulation engine is also the part of the research. This application is created by using Microsoft Visual C# programming environment, as the .NET framework supplies powerful libraries to handle the defined input sources and to implement the necessary input interfaces (with special attention to Excel sheets, which can be time-consuming to read).

Since this self-developed simulation application is created to serve as a native part of the EasySim system, therefore it is a well-tailored application, which operates more effectively in this case than a general-purpose commercial simulation system. Therefore enhanced performance (running speed) can be achieved.

4.2. Test case and validation

To demonstrate the capabilities of the developed system and to validate it the model of a theoretical reference production system was used. It is a Job-shop model, which represents the routings of 3 different products on 5 different workstations. At each workstation there are different distributions of working time for each product type. Figure 7 illustrates the model of the reference production system (which is based on the example described in [8] pp.155-157).

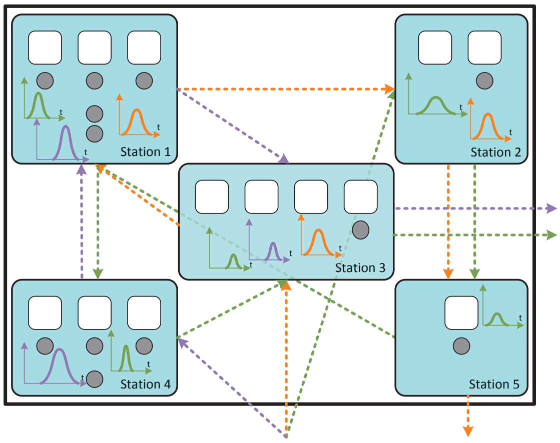

Fig. 7: The illustration of the implemented test-case. The colored lines denote the different production routings.

All the necessary data, which is required to represent this manufacturing system was fed into the uniform data structure Excel sheets. The defined KPIs were aimed to perform a bottleneck analysis, therefore Overall Equipment Efficiency (OEE) and average queue size and average waiting times were analyzed.

During the validation of the results, the use of both Plant Simulation and the self-developed simulation application enables to compare the results of the different simulation applications in order to validate the results of the self-developed application. This comparison was executed with deterministic parameters in order to avoid the statistical differences. It turned out that both simulation engines gave the same results, which were in align with the results in [8].

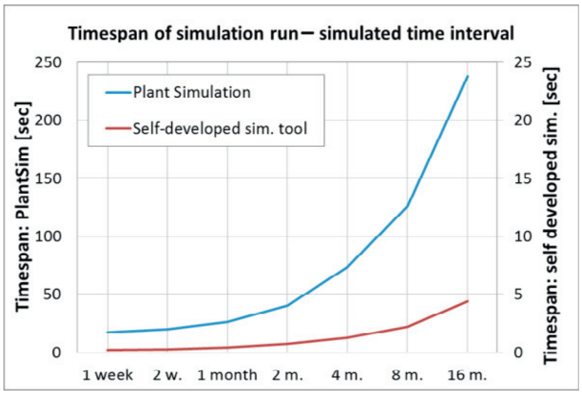

Fig. 8: Results of the comparison of simulation applications regarding the running speed of the test-case model.

In addition, during the validation a rough comparison of running speeds was also performed, which proved our presumptions that the more tailored self-developed application performs better than the general-purpose simulation product, in this special situation. The result of this comparison is shown in Figure 8. Simulation runs were executed with different simulation timespans.

To summarize the comparison of the commercial and self-developed simulation application, it can be stated that in cases, where limited functionality is acceptable (e.g. simple manufacturing systems) lower costs and better performance can be reached by applying a more tailored self-developed tool.

5. Conclusions

The main result of the research is the creation of the EasySim system, which supports the draft analysis of simple manufacturing structures. During the research the requirements of the system were specified and implemented, including the uniform data structure, the supporting user interfaces and in addition the self-developed simulation engine. The solution relies on the application of the uniform data structure, which was defined by following the guidelines of the ISA-95 international standard.

The uniform data structure allows modular system structure, which supports the application of different simulation engines, including the self-developed simulation engine. This offers a more tailored solution, which has therefore higher performance and lower costs, compared to the applied commercial, general purpose simulation application. The higher performance increases the number executable simulation cycles in a given timespan, which results in increased efficiency of the analysis.

6. Future works

Regarding the future works the first step is the implementation of numerous test-cases. In one hand it would be helpful in the further validation of the concept and the developed solutions, while on the other hand it would also support specifying the further extensions of the capabilities of the system.

As it was described above, the implementation of the user interface has a large impact on the overall usability of the system, therefore this could also be a place for further improvements. Improving the user interface would be the most helpful in handling more complex manufacturing systems.

Acknowledgements

The research has been partially supported by the National Office for Research and Technology (NKTH) grant "Digital, real-time enterprises and networks", OMFB-01638/2009. The research reported in this paper has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement No: NMP2 2010-228595, Virtual Factory Framework (VFF).

References

[1] Ayyagari, M., Small and Medium Enterprises Across the Globe, Small Business Enonomics 29, pp. 415-434
[2] Banks J., 1998. Handbook of Simulation, John Wiley & Sons, Inc.
[3] Csapó, K., 2009. The characteristics and development possibilities of fast-growing small and medium-sized companies in Hungary, PhD thesis, Corvinus University Budapest
[4] Fowler, J.W., Rose, O., 2004. Grand Challenges in Modeling and Simulation of Complex Manufacturing Systems, SIMULATION 2004 80, pp. 469-476.
[5] ISA, 2006, ISA-SP95 Enterprise-Control System Integration, http://www.isa.org
[6] Kádár, B., 2002. Intelligent approaches to manage changes and disturbances in manufacturing systems, PhD thesis, Budapest University of Technology and Economics.
[7] Kuehn W., 2006. Digital Factory – Simulation Enhancing the Product and Production Engineering Process. Proceedings of the Winter Simulation Conference, pp. 1899-1906.
[8] Law, A.M., Kelton, W.D., 2000. Simulation Modeling and Analysis, McGraw Hill, pp. 264-270.
[9] Page, B., Kreutzer, W., 2005. The Java Simulation Handbook, Shaker Verlag, pp. 12-21.
[10] Park, H.T., Kwak, J.G., Wand, G.N., Park, S.C., 2010. Plant model generation for PLC simulation. International Journal of Production Research 48, pp. 1517-1529.
[11] Popovics G., Pfeiffer A., Kádár B., Vén Z., Monostori L., 2011. PLC code processing for automatic simulation model generation, MAS 2011. 10th International conference on modeling and applied simulation, pp. 155-159.