

Discrete event and motion-oriented simulation for FMS

R. Carelli, W. Colombo

Instituto de Automatica - Universidad Nacional de San Juan (UNSJ)

San Martin 1109 (o), 5400 - San Juan, Argentina

Tel +54-64-213303, Fax +54-64-213672

E-mail: rcarelli@inaut.edu.ar

R. Bernhardt, G. Schreck

Fraunhofer Institute for Production Systems and Design Technology (IPK)

Director: Prof. Dr.h.c.mult. Dr.-Ing. G. Spur

Pascalstraße 8-9, D - 10587 Berlin, Germany

Tel +49-30-39006-0, Fax +49-30-3911037

E-mail: Gerhard.Schreck@ipk.fhg.d400.de

Abstract

In this work, an off-line simulation approach for simultaneously handling the discrete event and motion orient aspects of flexible manufacturing systems (FMS) is presented. A concept for signal handling in a discrete-event simulation based on Petri Nets and a proposed interface to motion simulation is outlined. The description of motion simulation covers the introduction of a unique interface concept for all FMS components. As an example for this approach, the RRS-Interface developed for robot simulation is used.

Keywords

Flexible manufacturing systems, discrete event simulation, Petri net, motion simulation

1 INTRODUCTION

Simulation techniques aid in optimizing flexible manufacturing system (FMS) design. A simulation program is produced in two stages (Figure 1). The first stage involves creating a formal model of the FMS. This entails abstracting physical reality as far as problem parameters allow.

The second stage converts this formal model into a simulation program with the aid of a simulation language. Using analysis of experiments with the simulation model (Figure 2), designers now seek evidence that the chosen formal model closely represents the actual system (validation) and that the simulation program satisfies the formal model (verification) [1].

Discrete simulation refers to the jumping by the simulation model time parameter from one event to the next; simulations without jumps are called continuous. Most FMS involve discrete simulation programs, although mixed forms are possible. A continuous process may activate a discrete event, for example: the end of a continuous heating process may trigger a request for a transfer trolley.

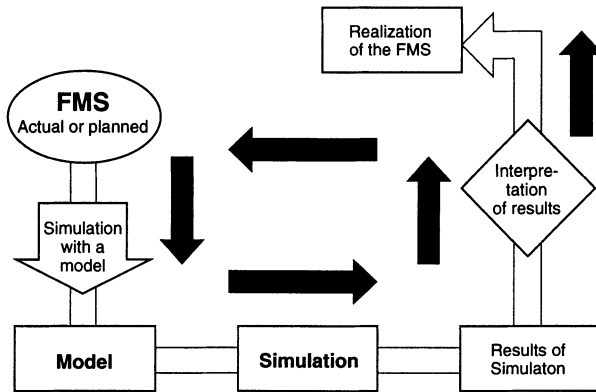


Figure 1 A formal model precedes each simulation program.

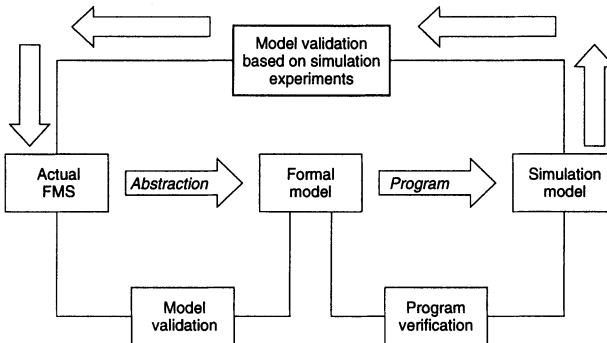


Figure 2 Designers realize a mature simulation only after several model-building stages.

An off-line simulation is used during the production planning phase and operates in isolation from the actual process. Numerous simulations, with widely varied data, must be run to obtain reliable information on future process behavior. It is also desirable, however, to optimize actual process control by coupling a simulation to an actual system in operation. This immediate integration of actual data is known as on-line simulation.

In this work an off-line simulation approach is proposed for handling the discrete and the motion oriented (continuous) aspects of the FMS operation simultaneously. This off-line simulation allows to dimension the systems under study and to determine process control parameters for satisfying -insofar as possible- the following objectives [2]:

- Maximum utilization of machines and personnel;
- minimum set-up times;
- short lead times for orders;
- minimum workshop and warehouse stockage levels;
- low transportation costs;
- investment optimization;
- other objectives may be added, depending on the applications.

Furthermore output variables are influenced by

- the work (machine sequence);
- the capacity of machines and personnel available;
- the batch size;
- the sequencing of incoming orders;
- the number and type of transportation facilities;
- the production process;
- future trends in the volume and range of orders.

At this stage, the simulation aims to investigate the effects of changing input parameters on the overall system, as well as to define the functional relationships between input parameters and output variables. By including investment considerations and other criteria, like flexibility and delivery times, analysts can assess -using this proposed methodology- the overall system and describe the best approach.

At present discrete event and motion oriented simulations are performed by using different modelling techniques implemented in separated simulation tools. The benefits of an integrated simulation environment, covering both aspects have been investigated in prototypical implementations at IPK. For a robotized FMS system a motion and process related simulation module has been connected to the high level cell control system. The development and test of cell programs (event level) has been supported by simulation and led to a reduction of set-up times at the physical system. However this experiments have been performed in a specific implementation dedicated to the available FMS system and the related system components (e.g. cell controller). In the frame of the cooperative work between INAUT and IPK a more generalised approach, independent of specific components is investigated. For this reason a discrete event simulation module (based on Petri Nets), under development at INAUT, will be used to cover the high level control and to identify the required interfaces to motion simulation modules.

2 SIGNAL HANDLING IN THE DISCRETE-EVENT SIMULATION MODEL

A manufacturing system -whether simple or complex- includes raw materials, tools, operators, and control policies for the operation of the equipments. At the sector level, it also includes flow of material and information through a collection of tools. It can even extend to the procurement,

production and distribution networks of suppliers, plants, and distributors. Using computer simulation for modelling and analyzing such systems helps the designers to predict and improve a system's performance, as measured by such elements as capacity, cycle time, inventory, utilization, service level, and costs [3].

Information about the dynamics and kinematics of the involved equipment of the FMS, the product routing, and the production and parts-availability schedules must be the inputs to the discrete event simulator. Using the graphical user interface of the simulator, these inputs could be described and modified easily without programming. Information such as the production capacity of machines on the FMS, the duration of various operations performed by different machines on different products, buffer sizes, and the staffing must be processed before they entry to the simulator, in order to allow their adaptability to the signal handling of the discrete-event simulator.

Key outputs of the discrete-event simulator are the system throughput (the quantity of products produced in each time period), a work-in-process inventory (the quantity of products being worked on in the system), the cycle time (the time from release of jobs into the system to completion), and machine utilization (the proportion of time the machines work on products). These results can be displayed as bar or line charts. By examining the above named results, the designers can identify easily the bottlenecks of the systems under study, and suggest on how to alleviate the problems that could be generated. Figure 3 shows a typical scheme about the functioning of a manufacturing simulator, and its more common input and output information [4].

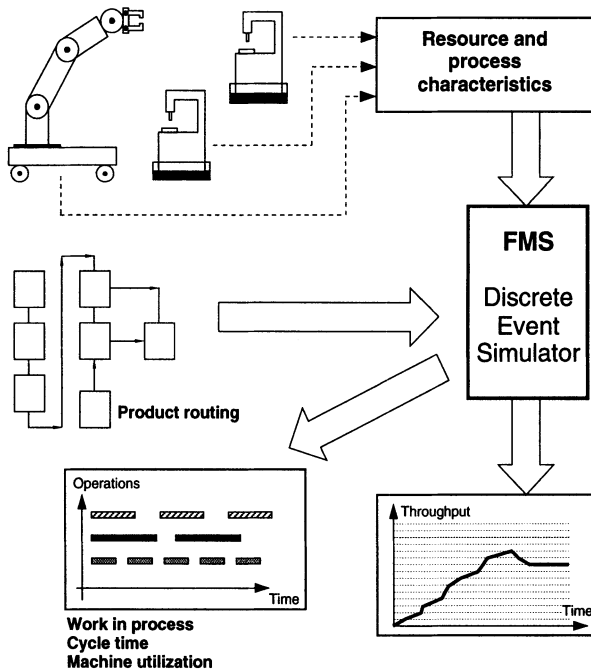


Figure 3 Input and output information related to a FMS discrete event simulator.

Since a simulation imitates the behavior of a system as it evolves over time, basic to the approach of this simulation is the building of a model that highlights the vital characteristics of the system [5]. Good models are needed to capture the characteristics of the discrete event systems [6], namely: concurrency or parallelism, asynchronous operations, deadlock, conflicts and event driven. In this work it is proposed the use of Petri Nets (PNs) as a discrete event specification tool for the modelling of the manufacturing systems under study. Petri Nets have been designed specifically to model FMSs [6], [7], [8]. They provide suitable models for the following reasons:

- PNs capture the precedence relations and structural interactions of unpredictable, concurrent, and asynchronous events. In addition, their graphical nature helps to visualize such complex FMSs.
- Deadlock, conflicts, and buffer sizes can be modelled easily and efficiently.
- PNs models represent a hierarchical modelling tool with a well-developed mathematical and graphical foundation.
- PNs allow the use of different abstraction level models and the use of powerful modelling methodologies: refinement and modular composition, each of them with its corresponding validation analysis.

It is very important to take into account that the main function of FMSs is to input rough and separate material parts, to achieve a certain amount of transformations, assemblies, disassemblies on the initial parts, and finally to output finished parts. Therefore, the designer has to identify the input and output sequences of material parts, at the same time all the elementary operations which have to be applied on these parts, and the main characteristics of the resources (turning, milling machines, manipulators, AGVs, etc.) involved on each of these operations. According to the above considerations, the Petri Net based discrete event simulator has a logical model where [9]:

- One place is associated to one state of the produced part in its operating sequence, or the state of a resource (idle or busy by a part).
- One transition is associated to only one operation on the produced part, or to the assembly operation of several parts.
- One token is associated to one material part, or information about the resources of the FMS.
- To each place of the net, a capacity place or capacity monitor is associated to indicate the maximum number of tokens this place may content. This capacity is introduced to indicate that a buffer, a storage area, or another resource can only contain a limited number of parts.

In addition, all the resources and operations verify the following set of constraints:

- The operations are completed in finite time.
- One part is submitted to only one operation at a time, and each resource can perform only one task at a time.
- One separate part can be submitted only to transformational or informational functions. A transformational function consists in modifying the physical attributes of the part (shape, constitution, surface, etc.). They are the machining functions: turning, milling, manipulation, and conditioning functions: termical treatment, painting, washing, etc.. An informational function consists in verifying that the operations have been accomplished correctly.

At the moment, a delay on transitions are usually used to model the dynamic behavior of the PN model. This temporization must be compatible with the capacity mechanism and with the control processes which are modelled. The discrete event simulator presented here has the parameter time associated to each temporized transition.

Petri Net models

Petri nets are now defined so that Petri nets can be discussed in a formal manner. The notation introduced in this section will be used throughout the paper. More details about this tool are in the following references [10], [11], [12].

Formal Definition

A Petri net (PN) is a directed graph consisting of two types of nodes, places and transitions, in which places are always followed by transitions, and vice versa. A Petri net is defined by

$$PN = \{P, T, E, S, m_0\}. \tag{1}$$

Where:

- P** : Set of the PN places. This set has cardinality *m*.
 - T** : Set of the PN transitions. This set has cardinality *n*.
 - E** : Input matrix. It can be seen like a function $E:P \times T \rightarrow N$
Each element of *E* represents the weight of the input arc from place *p_i* to transition *t_i*.
 - S** : Output matrix. It can be seen like a function $S:T \times P \rightarrow N$.
Each element of *S* represents the weight of the output arc from transition *t_i* to place *p_i*.
- m*₀ is the initial marking, with dim (*m*₀) = *m*.

A Temporized Petri Net is now defined by:

$$TPN = \{P, T, E, S, m_0, q\}. \tag{2}$$

Where each element has been previously defined, and *q* represents a time function. It attaches a time to each temporized transition of the net, that is $q : T \rightarrow R^+$.

The token-game of the *TPN* is performed as described in Figure 4. According to the adopted modelation methodology, a transition models an operation of T time units in the modelled system. The editor of the discrete event simulator represents the transitions with blue boxes. When a transition is enabled, the corresponding box becomes light blue color during a time T period (the modelled operation is actually performed). After the time T, the transition fires (the modelled operation finished). The last condition is showed by a green color box during a unit graphical time in the simulator.

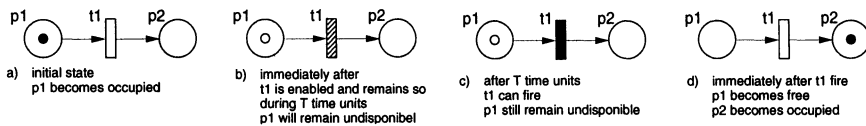


Figure 4 Token-game for a TPN model in the proposed discrete event simulator.

3 MOTION ORIENTED SIMULATION OF FMS

An abstract simulation of FMS can be done at the event level. A more precise simulation requires the consideration of causes of events. At the physical system, events are defined related to states of technological processes or motion states of FMS components. For the second aspect, motion models are required for all components which can generate a motion (e.g. industrial robot, conveyor, turn table). These components can be characterized as active components. Their drives can be controlled by numerical controllers including feedback loops or by simple on/off switches using a PLC. In the case of an industrial robot the motion model is represented by a model of the behavior of the robot controller (control model) [2]. This example will be used for further investigations.

Robot control models can be used to simulate the motion path, but also to calculate the motion time which is an important value for the estimation of cycle times. The problem arises in getting a control model which represents the real system behavior and which leads to precise simulation results. For this reason, a consortium of European automotive companies initiated the project Realistic Robot Simulation (RRS) [13]. The goal of the project was to overcome the inconvenience of accuracy by defining a simulation system interface, which enables the integration of control models delivered by the robot manufacturer. Furthermore, the interface concept had to allow the integration of original controller software. In the meantime the interface specification is available [14] and the RRS-Interface has been implemented in off-line-programming and simulation systems on the market. First robot control models are integrated and in industrial application.

The RRS-Interface had to be uniform for all controllers, but also had to cover the wide range of different concepts for motion handling. In the result, the interface provides a set of principle services. Without these services, no reasonable operation is possible. In addition, several groups of further services are defined (e.g. services for motion related event handling on RC level). The basic idea of the RRS approach is to use precise simulation models delivered by the control / robot manufacturer. For motion simulation, this concept could also be applied for other FMS components, if a suitable and accepted interface is available, which includes the definition of services to be provided by simulation modules. The usability of the RRS-Interface concept as a basic approach will be subject of further investigations.

4 INTEGRATION OF MOTION ORIENTED SIMULATION

From the control point of view, the discrete event simulator communicates with the real FMS or the motion oriented simulator through a set of signals such as: orders for the execution of activities, activity-end messages, sensor signals, alarm and error signals. In this work, the motion oriented simulator has replaced the real resources of the FMS and simulates the signal interchange with the discrete event simulator. It is necessary to develop a special system called event monitor, which is responsible for the interfacing between the PN model and the motion oriented simulator.

The interactions from the motion oriented simulator to the discrete event simulator are called events, and the interactions from the discrete event simulator to the motion oriented simulator are termed actions. A transition enabling and firing (according to Figure 4) may suppose the execution of an associated action. If this action implies an external action, a message is sent to the event monitor. Events are associated to the incoming signals from sensors or alarm signals

(such as machine and part breakdowns, abnormal stoppages, etc.). Events produce additional constraints for the transition firing rule of the PN based discrete event simulator.

During simulation time, the discrete event simulator makes the interpretation of the dynamic behavior specifications represented in the system model by the underlying Petri Net. The discrete event simulator performs the token-game in the net, which generates actions sent to the appropriate module of the motion oriented simulator (Figure 5). At the occurrence of an event, a message is sent back to the discrete simulator producing a new marking which enables new transitions.

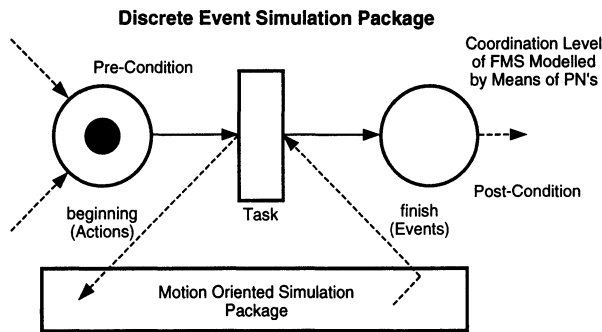


Figure 5 Interface for Communication between both Software Packages.

5 CONCLUSIONS

The present paper gives an idea of the main research goals of the joint project between the Instituto de Automática (INAUT) -Argentina, and the Fraunhofer Institut (IPK) -Germany, and the way they are associated to develop a whole simulation based methodology for designing and analyzing flexible manufacturing systems. Both research groups have decided to develop the concept of an hybrid software tool based on the integration of discrete event and motion oriented simulation concepts. The INAUT provides a recently developed Petri Net based discrete event simulator [4], and the IPK a motion oriented simulator. In this work, a basic interface architecture for connecting the above software tools is proposed. In a first step the FMS system from IPK has been modelled and simulated by using the Petri Net simulator. By associating a firing delay time to each operation modelled by means of the transitions of the Temporized Petri Nets, a Gantt diagram was obtained. When both simulators (discrete event and motion oriented) work together, the firing delay time will be obtained from the continuous simulation.

The obtained results are promising and shows that the proposed methodology allows to test different FMS configurations and to attain:

- quantitative information about the behavior of the simulated system;
- a fast prototype of the system under study;
- qualitative information on the system evolution.

6 ACKNOWLEDGEMENT

The work presented in this paper is the result of the cooperative action in the frame of the ECLA FlexSys project (partly funded by the European Commission) between the Instituto de Automatica San Juan, Argentina and the IPK Berlin, Germany.

7 REFERENCES

- [1] Willi Hardeck. "Simulation optimizes planning and design". Siemens Review 2/85. Vol. LII n° 2, p.p. 28-29. March/April 1985.
- [2] R. Bernhardt, G. Schreck, A. W. Colombo, R. Carelli. "Integrated Discrete Event and Motion Oriented Simulation for Flexible Manufacturing Systems". Journal on Studies in Informatics and Control. Vol. 3, n° 2-3, p.p. 209-217. Ed. Romanian Academy. September 1994.
- [3] Rangarajan Jayaraman (IBM Research Center). "Manufacturing Systems Simulated". IEEE Spectrum 9/93, p.p. 60-62. September 1993.
- [4] A. W. Colombo, J. Pellicer, M. Martín. B. Kuchen. "Simulador de Sistemas Flexibles de Manufactura usando Redes de Petri Temporizadas". Proceedings of 6° Congreso Latino Americano de Control Automático, vol. 1, p.p. 100-102. Río de Janeiro. Brasil. 19 al 23 de Septiembre de 1994.
- [5] J. Ezpeleta, J. Martínez. "Petri Nets as a Specification Language for Manufacturing Systems". 13th. IMACS World Congress on Computation and Applied Mathematics. Dublin. July 1991.
- [6] M. Silva, R. Valette. "Petri Nets and Flexible Manufacturing Systems". Advances in Petri Nets. Lecture Notes in Computer Sciences, vol. 424, p.p. 374-417. 1989.
- [7] Invited Sessions on Petri Nets and Flexible Manufacturing. Proceedings of the International IEEE Conference on Robotics and Automation. Vol. 2, p.p. 999-1186. Raleigh, North Carolina. EEUU. March/April 1987.
- [8] Proceedings of the 3rd International IEEE Conference on Computer Integrated Manufacturing. Rensselaer Polytechnic Institute. Troy, New York. EEUU. May 1992.
- [9] A. W. Colombo. "Modelling and Analysis of Flexible Production Systems". MsC Thesis. Universidad Nacional de San Juan. Argentina. November 1994.
- [10] J. L. Peterson. "Petri Net Theory and the Modelling of Systems". Prentice Hall. Inc. Englewood Cliffs. N.Y. 1981.
- [11] M. Silva. "Las Redes de Petri en la Automática y las Informática". Ed. AC, Madrid. España. 1985.
- [12] T. Murata. "Petri Nets: Properties, Analysis and Applications". Proceedings of the IEEE, vol. 77, n° 4, p.p. 541-580. April 1989.
- [13] R. Bernhardt, G. Schreck, C. Willnow. "The Realistic Robot Simulation (RRS) Interface". Proceedings of 2nd IFAC/IFIP/IFORS Workshop on Intelligent Manufacturing Systems (IMS '94). June 13-15, 1994, Vienna, Austria.
- [14] N.N. (1994) RRS-Interface Specification (Version 1.0), distributed via IPK-Berlin.

8 BIOGRAPHY

Ricardo Carelli received the Electrical Engineering degree (with honours) from the National University of San Juan (Argentina) in 1976 and the PhD degree in Electrical Engineering from the National University of Mexico (UNAM) in 1989. He is presently professor of Automatic Control at the National University of San Juan and research-worker of the National Council for Scientific and Technical Research (CONICET). His current research interests are in the fields of robot control, manufacturing systems and artificial intelligence applied to systems control. Dr. Carelli is a member of IEEE and AADECA, The Argentine NMO of IFAC.

Armando Walter Colombo was born in Mendoza, Argentina in 1960. He graduated as an Electronic Engineer from the National Polytechnic University of Mendoza in 1990. He received the Master Degree in Control Engineering at the National University of San Juan in 1994. He is currently pursuing his Doctoral Degree on Design of Flexible Production Systems and is developing a DAAD fellowship at the University of Erlangen-Nürnberg, Germany. His interests are in the fields of automatic control, Petri Nets and flexible manufacturing systems.

Rolf Bernhardt studied Electrotechnique and Communication Technique at the Fachhochschule Coburg and at the Technical University Berlin (TUB) at which he also received a Doctoral Degree in the field of navigation systems. Since 1984 he is working with the Fraunhofer Gesellschaft, Institute for Production Systems and Design Technology (IPK) in Berlin. He is and was involved in many national and international industrial and public funded R&TD projects in the area of planning, simulation and programming of manufacturing systems with robots. At present he is head of department in the division Control Techniques for Production Systems and provisional director for R&TD in robotics. He is lecturer for Automation at the TUB.

Gerhard Schreck studied Mechanical Engineering at the Technical University Berlin. Since 1985 he is a researcher at the Fraunhofer Gesellschaft, Institute for Production Systems and Design Technology (IPK, Berlin) in the area of off-line programming and simulation for industrial robots and was involved in many national and international projects for robot application in manufacturing, off-shore, and space. At present he is head of department in the division Control Techniques for Production Systems at IPK.