

ENTERPRISE SCHEDULING: HYBRID AND HIERARCHICAL ISSUES

John Pastrana

University of Central Florida
4000 Central Florida Blvd.
Orlando, FL 32816, USA

Mario Marin

University of Central Florida
4000 Central Florida Blvd.
Orlando, FL 32816, USA

Magdy Helal

Benha University
Industrial Engineering
EGYPT

Carlos Mendizabal

Universidad Technologica de Panama
Campus Universitario Victor Levi
Panama City, PANAMA

ABSTRACT

We build a hybrid discrete-continuous simulation model of the manufacturing enterprise system. This model consists of an overall system dynamics model of the manufacturing enterprise and connected to it are a number of discrete event simulations for selected operational and tactical functions. System dynamics modeling best fits the macroscopic nature of activities at the higher management levels while the discrete models best fit the microscopic nature of the operational and tactical levels. An advanced mechanism based on information theory is used for the integration of the different simulation modeling modalities. In addition, the impact of the decisions at the factory level in scheduling are analyzed at the management level. The different models of control are discussed.

1 INTRODUCTION

The advances in information and computing technologies in addition to the unprecedented levels of competition and fast pace of changes in the business environment have created a more flattened enterprise system and changed the way enterprises should be managed. This is creating challenges to using simulation tools. The presence of manufacturing and nonmanufacturing functions in the manufacturing enterprise, the different types of behavior in the system, differences between management levels in the scope of planning, frequency of decision making, and levels of details they deal with, indicate that a single simulation approach cannot offer all that is needed in a simulation of such a complex system.

The traditional use of discrete event simulation (DES) to simulate the manufacturing systems has narrowed the scope to detailed statistical analyses at the operational levels of the system. At that level, the main concerns have been the flow of materials and the levels of inventories and work in process, and the performance at the individual level of resources, units of products or processes (Smith 2003; Lee et al. 2002; Law and Kelton 2000; De Souza et al. 1996; Pegden, Shannon, and Sadowski 1990; Carrie 1988). The need to simulate the whole system (aggregate and detailed management level functions) has challenged DES, which was inadequate to approximate the continuous behavior in the system generally and particularly at the aggregate levels, or communicate appropriately the financial computations to higher management (Lee et al. 2002; Johnson and Eberlein 2002; Barton, Love, and Taylor 2001). DES works effectively with problems of narrow scopes, but it is "... incompatible with a global point of view" (Lin et al. 1998). And it does not address the stability of the system (Rabelo et al. 2005) that should be consi-

dered at an aggregate level before any detailed analyses can be conducted (Towill and Edghill 1989). Add to that its high demand for data and its tedious data preparation processes. There are always detailed data available for the manufacturing functions. But for the aggregate management levels data is not usually available and in most cases only rough estimates exist (Zulch, Jonsson, and Fischer 2002; Mandal and Sohal, 1998; Anthony, Dearden, and Bedford 1989).

Meanwhile, SD has been successful as a system thinking approach that targets top management levels with a comprehensive integrative perspective, with relatively minimal data requirements. SD is appropriate for modeling large scale systems and the higher levels of decision making where aggregation is preferred. It focuses on the policy decisions that are embodied in the feedback loops, and not on individual localized decisions. Nevertheless, using SD at the operational level of the manufacturing system has failed to offer the needed granularity (Godding, Sarjoughian, and Kempf 2003; Barton, Love, and Taylor 2001; Baines and Harisson 1999; Bauer, Whitehouse, and Brooks 1982). The same was observed by Choi, Bae, and Kim (2006) who could not use SD to model the performance of the individual processes in a software development system. In addition, while SD permits the study of the stability of the system over the long range, the trends of behavior that it generates do not indicate what specific actions to be made and at what values of the action parameters. Such specificity requires more detailed considerations that SD does not seem to work with, while DES has been effective at.

Consequently, we have argued (Rabelo et al. 2005; Helal and Rabelo 2004) that SD and DES can complement each other to offer the needed tools to meet the needs created by the modern, integrated manufacturing enterprise system. In this paper, we describe the details of our proposed SDDDES simulation method.

2 HYBRID DISCRETE-CONTINUOUS MODELING

2.1 SDDDES Simulation Approach

A combination of SD and DES simulation paradigms has the potentials of satisfying the needed characteristics in the simulation model of the integrated manufacturing enterprise. The integration of SD and DES as proposed in this work offers an inexpensive technique that maintains the existing simulation expertise in simulating the manufacturing systems. Legacy system models can be utilized and no new programming skills are needed to use SDDDES.

Figure 1 depicts a roadmap in which the review of current practices in using simulation approaches has indicated the potentials of integrating SD and DES. SDDDES is a hybrid continuous-discrete method for simulating the manufacturing enterprise. The size of the simulation model suggested following a modular structure. Modules should be formally described for better model management and communication among modelers. The SDDDES formalism is proposed for that purpose. And since SDDDES combines two different simulation paradigms, it should also be viewed from the perspective of a distributed simulation arrangement. Specifically, the SDDDES synchronization mechanism is proposed to coordinate and synchronize the interactions among the SD and DES simulation modules. The SDDDES controller is the main unit in the SDDDES simulation method. It manages the integration, implements the synchronization, and offers the user interface.

SDDDES consists of an overall SD model for the manufacturing enterprise system and a number of DES models built for selected units in the system as dictated by the analysis needs. The models will interact through the SDDDES controller.

Main use of the SDDDES models is for management to test policies before deciding on implementing them, to confirm the estimated behavior based on enterprise-wide feedback. Management would use the model to investigate the feasibility and desirability of the various programs and initiations in the organization as well as use it to test the resource allocation process to decide on the strategic plans. The feedback structure in the model, in addition, helps better understand the dynamics of the interactions among the system components. In addition, the model can be used as a comprehensive performance measurement

system and this can be accomplished by building a balanced scorecard at the top of the model components and parameters. Helal and Rabelo (2004) have discussed the potentials of building dynamic balanced score cards based on SDDDES.

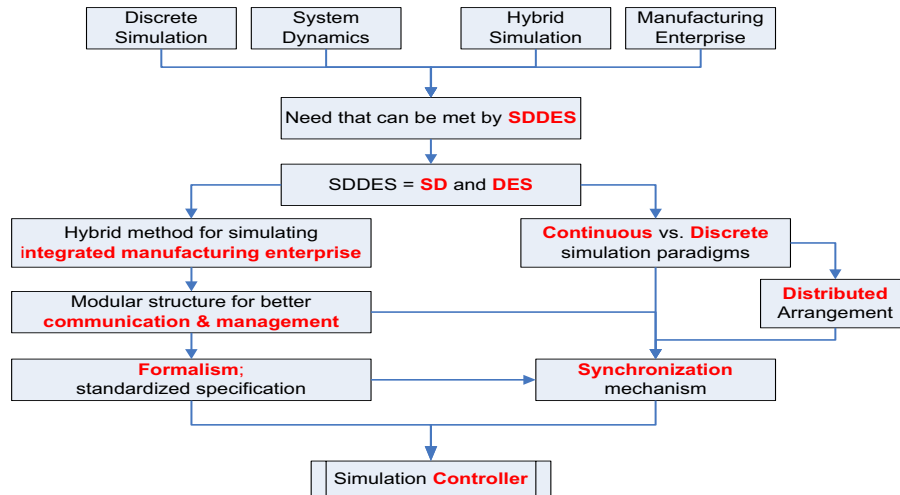


Figure 1: Road map toward SDDDES

2.2 The SDDDES Modules

Modules facilitate the model building process, especially when modeling the complex structure of the manufacturing enterprise. They also simplify modifying and extending the model when desirable. Each module has a well defined function, which should have a set of inputs and outputs to be ready to interact with other. The stock management model (Sterman, 1989, 2000) offers a good basis to modularize SD models. DES modules are defined based on the functions they will perform and built from scratch if they do not already exist.

2.2.1 The SD Modules

If SD models already exist then they can be modularized such that they are based on the stock management model. New models can utilize the stock management model. We recommend, but it is necessary in SDDDES, the use of the stock management model to standardize the building of SD models. Sterman (2000) listed and explained numerous examples for using the stock management model in business systems and other types of systems. Modularizing is an iterative process (Figure 2). The outcome of the process is a set of modules with the inputs and outputs that are to be exchanged between them so they function correctly. The defining of modules should be simultaneous for all SD modules and the DES modules, if necessary. It should be noted that the modularized SD model is still a single SD model. Modules are logical for the purposes of integration with the DES models and for communication and model management uses. SD modules are treated as sections in the SD model, yet are formalized (using the SDDDES formalism) and defined at the SDDDES controller as separate units. Not actually dividing the SD model maintains the integrity of the feedback structure of SD while simplifying working with the model in SDDDES.

The module represents a function. The inputs to a module are variables that are not normally under the control of the unit manager, or normally included in the core definition of that function represented in the module. To recognize the interactions that unit would have with other units the definition of it could need to be modified based on the requirements of the other modules, following the iterative process of Figure 2.

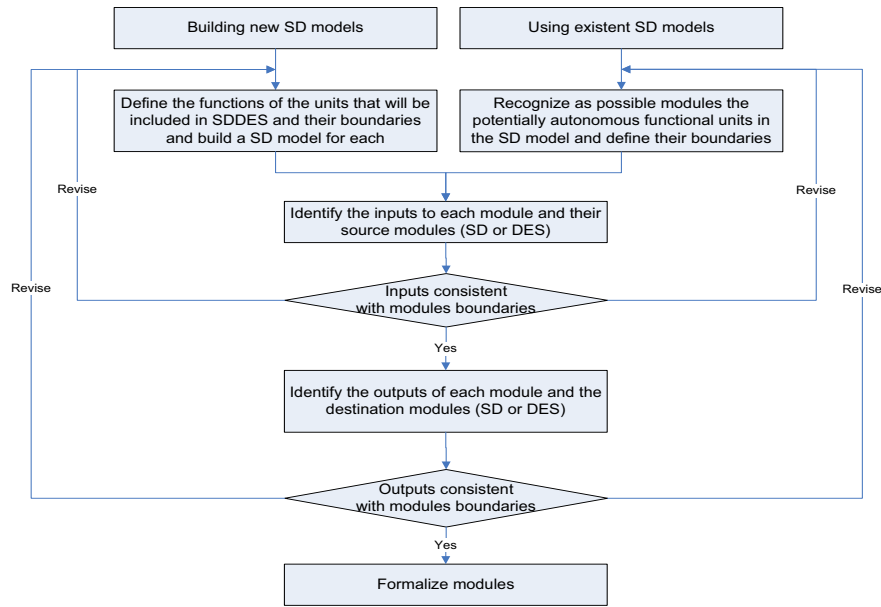


Figure 2: Developing SD Modules

The generic SD module can be represented as in Figure 3. Inside the module is a stock management model that is a model of the function of the module. Not all details need to be shown. Yet the sets of inputs and outputs must be well defined. The *In_Port_x* represents the input ports of communication where the module receives inputs from other SD or DES modules (*x* represents the number of the port). These are connected to the appropriate variables in the model. The *Out_Port_x* represents the output ports of communication where the module offers outputs to other modules. Communications through the ports are managed and synchronized by the SDDES controller. Defining the inputs and outputs is part of the module definition and they are used in the formalism of the module.

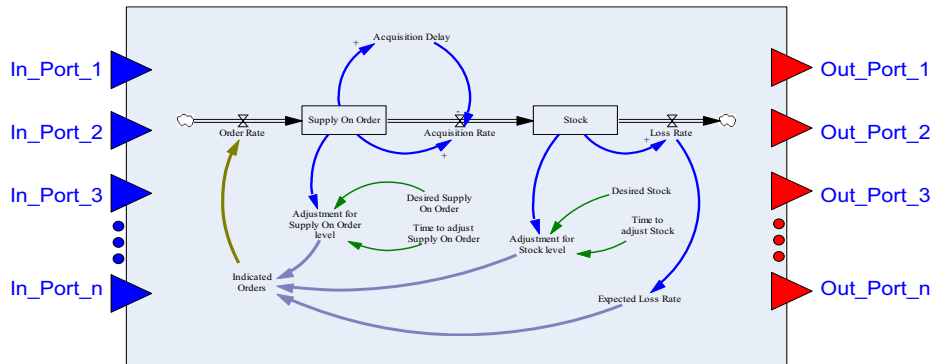


Figure 3: Generic SD module for SDDES

2.2.2 The DES modules

The role of the DES modules in the SDDES is to provide the needed detailed analyses. SDDES allows the use of small-sized DES models instead of the usual practice of building one large DES model of all production functions. This greatly cuts the model development time and expenses. The DES modules are built to interact with one or more SD modules or with each other. The DES modules are complete discrete simulation models that are of narrow scopes. An iterative process to develop DES modules is shown in Figure 4. It starts with a valid DES model for the function of concern. Inputs and outputs and modules to

interact with are then identified then the modules are described using the SDDDES formalism. Figure 5 shows a representation of a generic DES module. Any DES tool can be used. In this work Arena (Rockwell Automation 2010) is used in building DES modules while Vensim (Ventana Systems 2010) is used for the SD part.

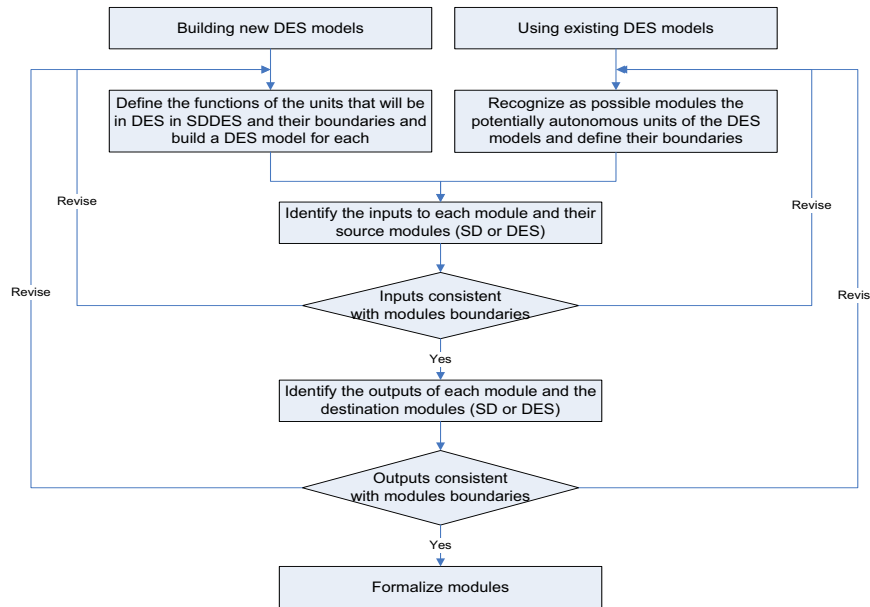


Figure 4: Module development for DES Modules

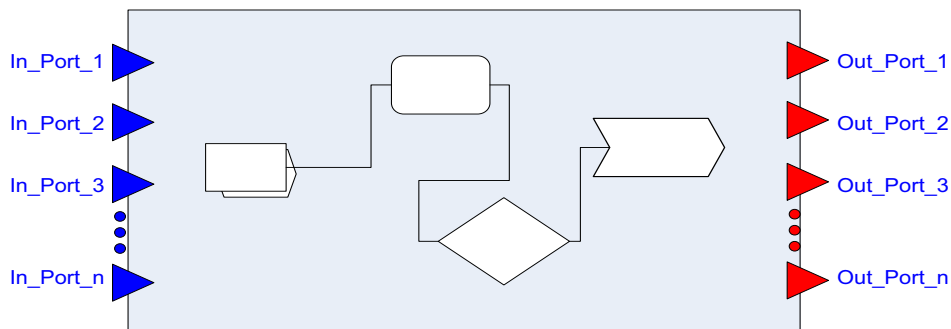


Figure 5: Generic DES module for SDDDES

2.3 SDDDES Controller

SDDDES uses the existing SD and DES modeling techniques as they are normally used. The integration of the modules and what it entails are all managed by the SDDDES controller. The SDDDES controller is the manager of synchronization of the SD and DES simulation modules in the SDDDES framework. The controller is a separate unit that interacts with the simulation modules to integrate them and facilitate the interactions between them according to the specifications included in the SDDDES formalism. In fact the formalism information is stored in the controller model database. The controller also implements the synchronization mechanism to control the running and stopping of the modules. It is also the user interface to perform I/O operations and to define/modify/replace the modules. Specifically, the SDDDES controller acts in the following areas:

- *Data management:* The controller ensures that the information indicated in the definition of the SD and DES modules (in their formalism specifications) are executed properly, in relation to the formatting of the data.

- *Time management:* The controller implements the synchronization mechanism to control the running of the modules. It monitors the simulation time. The DES modules do not run for the entire SDDDES simulation horizon in a single run, and the simulation times from them are not usable directly. The controller estimates the time for each with respect to the overall SD model such that the user can observe the correct time.
- *Participation management:* The controller offers the functionality needed to add new modules to the SDDDES model as well as to modify or replace already functioning modules. This is achieved through a user interface through which the modeler inputs the necessary data by the SDDDES formalism to specify a module.

All functions the controller does are based on the user interactions with its user interfaces. An IDEF0 of the SDDDES controller functions and details about how they are performed are presented in the following section.

2.4 Functional model of the SDDDES controller

The IDEF0 method offers a hierarchal representation of a system that depicts the functions done within the system along with relevant inputs needed to perform the functions, outputs generated upon performing the functions, the controls that guide and constrain the functions, and the mechanisms needed in that. The basic model is shown in Figure 6 presented from the point of view of the modeler/the user of the SDDDES. The A-0 IDEF0 model is the most abstract representation. A single box is used to indicate the function of the controller, namely Execute SDDDES. The SDDDES controller fundamentally executes the simulation run that the SDDDES model is built to make. The sets of inputs, controls, outputs, and mechanisms (the ICOMs) used in the A-0 model are described in Table 1.

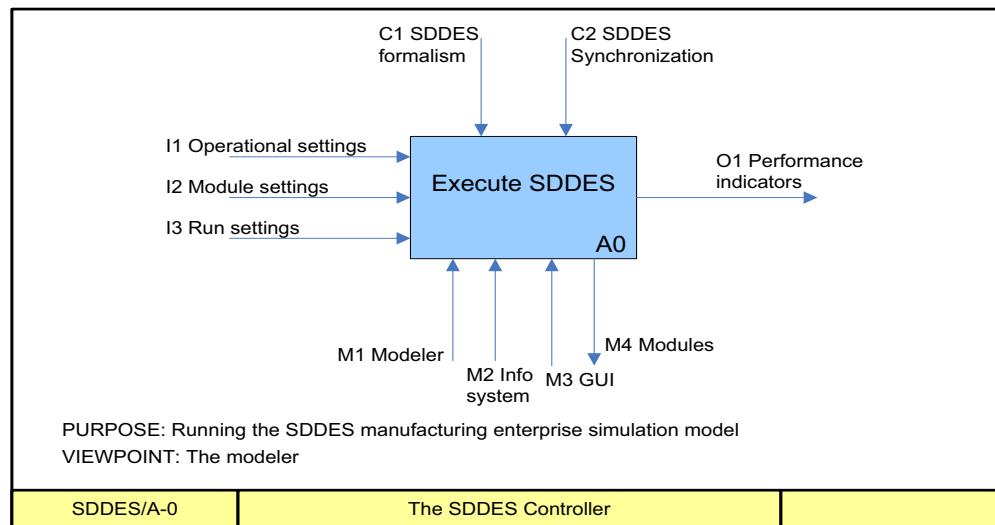


Figure 6: A-0 IDEF0 functional model of the SDDDES controller

The controller executes the SDDDES simulation model using these sets of inputs, controls, outputs, and mechanisms. The modeler uses the appropriate user interface to input module data or the simulation data. The modeler also observes the outputs during the simulation run and can pause the model to modify some settings or stop the simulation for a new experiment. The inputs provided by the modeler are specified by the two controls; the formalism and the synchronization algorithm. The controller contains a database to store the input data and the ongoing outputs during the run. To set the modules, the controller extracts the parameters of the modules (via calling M4) such that the modeler would assign values to them or link their values to the appropriate data in the information system. The decision on the TB for the DES modules can also use inputs from the information system.

Table 1: ICOMs for the A-0 IDEF0 model of the SDDDES controller

Inputs	I1	Operational settings	Characteristic information representing the current status of the system. They are elements of the management policies that will be tested and evaluated with the simulation model. The module parameters are assigned values in this action. These values are provided by the modeler or obtained from the active information system (M2).
	I2	Modules settings	The inputting of the data required by the SDDDES formalism. Modules can be modified, deleted from or added to the model.
	I3	Run settings	Specifying the planning horizon, number of replications for the DES modules, as well as the time units and needed parameters that will be monitored. Also the outputs that are of interest are specified here
Controls	C1	SDDDES formalism	This guides the addition, modification, or deletion of modules. Also specifies the data needed to set the model and the run.
	C2	SDDDES synchronization	This is SDDDES synchronization algorithm. It guides the simulation run and the data exchange transactions.
Outputs	O1	Performance indicators	This is the regular outputs of a simulation model
Mechanisms	M1	Modeler	Represents the user of the simulation model in general. The modeler performs all I/O operations
	M2	Info system	This is the existing information system of the company (e.g. ERP or MRP). Module variables are linked to data provided by the information system. Outputs can also be added to the information system.
	M3	GUI	The graphical user interface is an integrated unit of the controller. It offers several user interfaces through which the modeler interacts with the controller and the model.
	M4	Modules	These are the module information saved in their files (e.g. the Arena and Vensim files in the current work). They are called to be used as necessary by the modeler and during the run for sure.

The A-0 diagram is decomposed into more detailed definition of the controller functions. The A0 diagram of the IDEF0 model is the first level of details of the function described in the A-0 model. A0 for the SDDDES controller models the three basic functions of the controller as described in the previous section; described however in more practical terms. In the IDEF0 terms these functions are the A1, A2, and A3 in Figure 7. Each of these functions is decomposed further as necessary to offer a complete description of the controller role in the SDDDES model, prior to its implementation. The A0 model is described in Figure 7 and explained afterward.

The Interact With User function (A1) allows the user (the Modeler in the above model) to perform I/O operations as well as defining the modules. The appropriate GUI is initiated for the Modeler to input the necessary settings. These inputs are communicated to A2 and A3 for the models to be defined and the run to be ready to be executed. The GUI is developed to meet all use cases of the system and these use cases are controlled by the SDDDES formalism (adding or deleting modules), by the current contents of the saved modules (coming from A3 to modify modules, assign input values to their variables, etc.), and by the performance indicators (coming from A2 for the user to observe outputs and do necessary adjustments when desirable).

The Manage Model function works to accept changes in the existing modules and add new ones to the SDDDES model as inputted by the modeler in A1. The modules are saved in their simulation software files and the files are called as necessary (M4). The current contents of the modules are the outputs of A3 that are fed back to A2 so that the controller reads the time bucket setting for the modules and the

defined data exchange transactions that will be executed during the run in the A2 function. The synchronization algorithm (C2) controls A2 along with the relevant information from the formalism (C1). The current module contents from A3 are also fed back into A1 for the Modeler to correctly assign the operational and run settings.

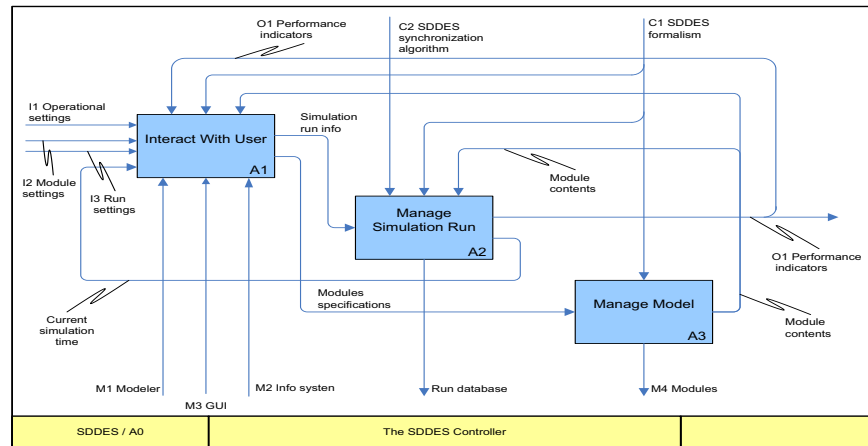


Figure 7: The A0 IDEF0 model of the SDDES controller

The output of A2 is the output of the simulation run, which is offered as the overall output of SDDES and is fed back to A1 for the modeler to analyze the performance with the appropriate GUI. It is noted that the A2 function is internal; no direct user interactions are needed with it. During the run, the behavior of the system is fed back to A1 for the user to perform any adjustment if desirable.

The output of A1 is the simulation run information representing the settings needed to start the simulation run at A2. These ongoing outputs are saved in the run database. They are updated all the time during the run. Of particular importance, the results of the run segments of the DES modules are saved to be used in the following segments.

3 APPLICATION OF SDDES TO A MANUFACTURING SYSTEM

3.1 Case Study

A case in which an SDDES hybrid simulation model was built for a real manufacturing system was used to describe the implementation and use of the concepts and elements of the SDDES framework. SD and DES modules were built to represent the manufacturing system of a eastern US based optical product manufacturer named Precision Molding Optics Corporation (refer herein as PMO Corp). The company offers its products to various markets including industrial, medical, defense, test and measurements, and communications markets. The precision molded optics (PMO) process produces lenses for industrial laser and other optical applications and is the focus of the simulation model.

Rapid technological advances and growth in the optical market (at the industrial and consumer electronics levels) challenged PMO Corp strategy of relying on providing customized products to loyal customers. Customers renovating optical systems were considering switching to competitors who can offer similar variety and quality at cheaper prices. PMO Corp main concern was the high scrap rate (estimated 25% to 40%). Scrap had been observed to be slightly increasing and as equipment gets older; it was projected to be higher. Higher scrap is translated into lower yields. Meaning, higher cost of production to cover customer demands was a driving force in the increase of product price and customer dissatisfaction, when product could not be delivered on-time. Recognizing the limits of its production capacity and technology PMO Corp was considering outsourcing the proven high demand products for cost reduction and not invest in new technology for their PMO production line.

3.2 PMO Corp Production

The PMO's production consisted of two main operations, the production of PreForms and use of Presses to form the final optical products. The PreForms and Presses departments contain all the production equipment. In PreForms, raw materials (glass slabs) are converted into semi-finished products called preforms (glass balls). Raw materials arrive in the form of slabs of 4 by 6 inches and 1 inch thick. Slabs are inspected for physical defects and thermal expansion characteristics and are then sliced into plates of different thicknesses to meet the different sizes of the various types of final product. Plates are formed into glass balls (preforms) of different sizes. Preforms are then annealed, ground, lapped, polished, smoked and baked, and cleaned before they are sent to the Presses department for the finishing operations. In Presses, the preforms are pressed on two types of presses to make usable lenses of different optical characteristics.

The PreForms department uses a relatively old technology. And because of the different types of final product, processing times and batch sizes at each processing step vary significantly. There is a single annealing oven that can process a batch of 1000 to 1500 units (depending on the size of the raw preforms) per run and it runs for an average of 12 hours. There are four grinding, five lapping, and 15 polishing machines that each can process between 100 and 350 units each run where the processing time vary from 25 minutes to more than five hours. The smoke and back and wash operations use heat and chemicals to clean the preforms in batches of up to 100 units and processing times are measured in minutes. The different lens sizes and necessarily the differences in batch sizes and processing times with the lack of computer control make synchronization critical to ensure the availability of materials to meet the Presses department requirements.

The Presses department is run as a pull production system that pulls raw preforms from PreForms according to the schedule to meet customer orders. Orders arrive in varying sizes for the different types of lenses. Raw preforms are pressed into final usable lenses at two types of pressing workstations: the air presses and the Nitrogen (Ni) presses. Each air press has seven pressing heads, which can be run separately. Each Ni press has six pressing heads, which are also independent from each other. Air or Ni refers to the technology used in the press where pressing is carried in a vacuum or within the presence of the inert Ni gas. Each type is capable of producing certain set of optical characteristics that overlap such that some types of lenses can be pressed on either type of press. Each press is run by a single operator. There are nine and seven Air and Ni presses respectively. After pressing, the operator inspects the lenses before they are forwarded to the more advanced interferometer testing station where they are accepted or scrapped. Operators retrieve the preforms daily according to the scheduled production of the day. Scrap rate is high at Pressing; ranging between 25% and 35% of attempted pressing operations.

3.3 PMO Corp SDDDES Model

The objective of the model was to demonstrate the implementation of the SDDDES simulation framework and its potential for modeling the manufacturing system and explaining its behavior and level of performance. In the case of PMO Corp model we needed to analyzed operations and evaluate if the PMO outsourcing option can be entertained or if there was a production or management problem. Thus, SD and DES simulation models were built to be used as modules in the SDDDES model. The SD and DES modules were built as regular SD or DES simulation models. Figure 8 shows the SD model of the PMO process internal supply chain.

The SDDDES hybrid simulation model of the PMO Corp precision molding operations includes the following units (Figure 9) to cover the operational level for the PMO process at the shop floor and the aggregate decision making level in addition to the financial and accounting aspects:

- The shop floor operations at the PreForms and Presses departments
- The internal supply chain represented by Materials Ordering, Production Planning, Inventory
- Management, and Shipping
- Labor Management

- Finance and accounting

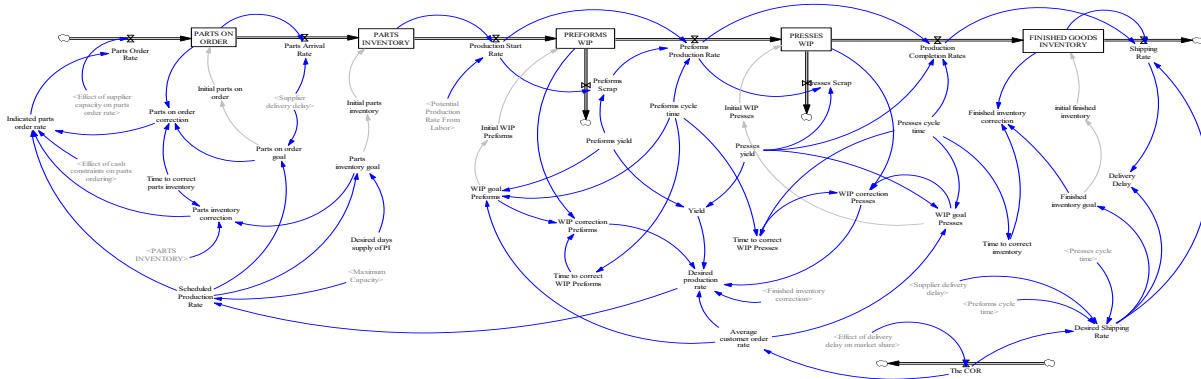


Figure 8: SD model of PMO Corp internal supply chain

It was shown that the SD and DES modules in the PMOC SDDDES model are interacting appropriately and the model could be used to explain the behavior of the system at the shop floor and planning level. It is concluded that the variation in production rate, particularly at the PreForms department and at the Presses department are the root cause of the oscillating behavior (although around a horizontal trend) observed for some of the system variables (inventory levels) and shipping performance. Even with a stable level of demand and no special occurrences at the shop floor, the system occasionally fell short of meeting demand and the inventory levels oscillated.

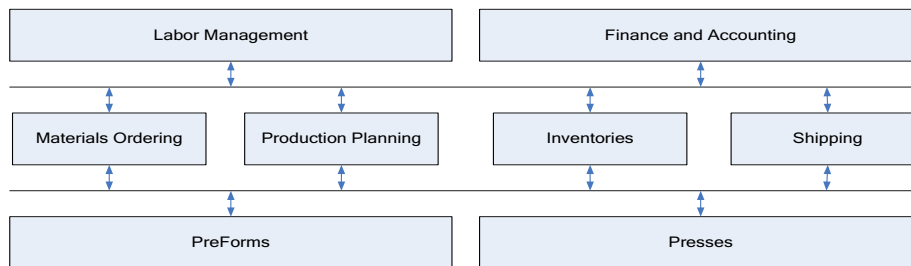


Figure 9: Business Units included in the SDDDES simulation model for PMO Corp.

We reviewed the model results with the PreForms department’s production manager and with the manager of operations of the company. With the fixed customer order rate they have recognized the behavior generated by the model as familiar adding that the variability in the daily production is due to the large number of product types with relatively small order sizes for each type. Their commitment to meet loyal customer orders led to the need to manage the production line in that manner. Synchronization between PreForms and Presses was continuously in focus but too tedious to achieve. The high scrap rate in the Presses would negatively affect the scheduling and synchronization efforts. Attracting larger order sizes and minimizing the number of variants of the product types was not an easy decision because of using pressing technology that was not economically competitive. The metallic molds for the pressing heads at the Presses department become defective when a pressing attempt fails. Molds would require to be sent for machining or they would be scrapped as well. Japanese and Korean and other competitors use ceramic-based pressing heads that had longer service lives and were cheaper. In addition, the company was committed to its long-term customer needs and its higher prices could not attract more demand. Unwilling to invest in renovating the production facilities at PMO, the company eventually outsourced the PMO operations in favor of directing their financial resources to the GRADUIM alloys production and other markets.

As for using the SDDDES model, it was perceived as a useful tool to improve the communications between the shop floor managers and the operations manager (VP level). The lack of timely communications and updated data between the operational and aggregate levels is a real concern. The estimates of the system capacity from the production manager and the operations manager (as expressed by the production manager) used to differ. The operations manager level uses data from the company's central database (Spreadsheet based application) and from other units including engineering, materials, etc. As described by the PreForms production manager:

- More often than otherwise, the VP team over estimates the system performance; the reason being the use of data from the centralized database as well as their most recent knowledge of the system capabilities. The production manager would request permits for using overtime or extra materials ordering (which are the VP decisions) to match the expectations.
- Due to significant differences in estimating the system yield (lower yield expected by VP than what the production manager knew about her production line capabilities!), they launched a project to evaluate each individual equipment unit's performance to update the central database. Need for that was caused by changes at the shop floor made by the production manager recently after she started her job. The data available at the VP level and in the data base was compiled under a previous production manager and could not be updated in a timely way.
- The production manager expressed an interest in having a tool that she could manipulate to reflect the current status (as it frequently changes) so she can communicate accurately to the VP and CEO and avoid the "frustration of convincing them" of the actual situation in the meetings.

The operations VP considered SDDDES valuable to experiment and evaluate the performance of the system and the impact of the resource allocation (materials and manpower) decision on the shipping performance. The causal relationships captured in SDDDES can save the time analyzing and summarizing the collected data at the central database. And he suggested including the database as an integral component of SDDDES to benefit from SDDDES in analyzing the system data.

4 SCHEDULING AND CONTROL

The stated purpose of using SDDDES is the management policy design and testing. As a simulation model, the uses of the SDDDES model are not limited to policy design. The control function is the first that should be mentioned. The common uses of simulation in the manufacturing system control applications are focused at the operational/shop floor level and are generally concerned with the ability to monitor the system on the short term basis (commonly in real-time) and the scheduling/rescheduling situations in response to uncertainties at the schedule execution levels (Jones, Yih, and Wallace 2001; Cowling and Johansson 2002; Aytug et al 2005; Son and Venkateswarana 2005; Cho, Son, and Jones 2006; Sinreich and Shints 2006; Pfeiffer et al. 2008).

SDDDES has been designed such that it maintains the integrity of the SD and DES simulation methodologies. Thus, in addition to the ability of the SDDDES to develop comprehensive simulations of the manufacturing system and explain its behavior, the feedback loops in SDDDES can provide the tools to assess the ability of the system to perform self-control and act to handle the deviations from the desired levels of behavior due to unexpected events at the operational or at the aggregate level as well. This can indicate if the system is adequately robust and well-structured.

5 CONCLUSION

This SDDDES simulation framework can build inexpensive, comprehensive, scalable simulations that support the management policy design and analysis on an enterprise-wide level. Simulation modeling has been and will continue to be the most effective way for analyzing the performance and designs of the manufacturing systems. But the evolution of the manufacturing system and the continual changes in the business environment have created serious challenges for the use of simulation in that area.

Our research asserts that the simulation of modern manufacturing systems should be approached using hybrid tools that can incorporate, in the same simulation, the operational and the aggregate management levels in a dynamic feedback-based structure. They need to be comprehensive in scope, scalable, and able to accommodate the continuous and discrete modes of behavior, the stochastic and deterministic natures of the various business units, and the detail complexity and dynamic complexity perspectives in the decision making processes. They also need to be practical to build and inexpensive.

REFERENCES

- Anthony, R., J. Dearden, and N. Bedford. 1989. *Management Control Systems*. 6th Ed. Irwin, IL.
- Aytug, H., M. Lawley, K. McKay, S. Mohan, and R. Uzsoy. 2005. Executing production schedules in the face of uncertainty: a review and some future directions. *European Journal of Operational Research* 161:86-110
- Baines, T., and D. Harrison. 1999. An opportunity for system dynamics in manufacturing system modeling. *Production Planning and Control* 10(6): 542-552.
- Barton, J., D. Love, and G. Taylor. 2001. Evaluating design implementation strategies using simulation. *International Journal of Production Economics* 72: 285-299.
- Bauer, C., G. Whitehouse, and G. Brooks. 1982. Computer simulation of production system: Phase I. *Technical Report COE No. 82-83-1*. The University of Central Florida, Orlando, FL.
- Carrie, A. 1988. *Simulation of manufacturing systems*. John Wiley & Sons, GB.
- Cho, H., T. Son, and A. Jones. 2006. Design and conceptual development of shop-floor controllers through the manipulation of process plans. *International Journal of Computer Integrated Manufacturing* 19(4):359-376.
- Choi, K., D. Bae, and T. Kim. 2006. An approach to a hybrid software process simulation using the DEVS formalism. *Software Process: Improvement and Practice* 11(4):373-383.
- Cowling, P., and M. Johansson. 2002. Using real time information for effective dynamic scheduling. *European Journal of Operational Research* (139):230-244.
- De-Souza, R., R. Huynh, M. Chandrashekar, and D. Thevenard. 1996. A comparison of modeling paradigms for manufacturing line. *IEEE Int. conf on systems, management, and cybernetics*, Oct 14-17, Beijing, China.
- Godding, G., H. Sarjoughian, and K. Kempf. 2003. Semiconductor supply network simulation. In *Proceedings of 2003 Winter Simulation Conference*, ed. S. E. Chick, P. J. Sanchez, D. M. Ferrin, and D. J. Morrice, 1593-1601. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.
- Helal, M., and L. Rabelo. 2004. An enterprise simulation approach to the development of dynamic balanced scorecards. In *Proceeding of American Society of Engineering Management Conference (ASEM'04)*, Oct 20-23, Alexandria, Virginia
- Johnson, S., and B. Eberlein. 2002. Alternative modeling approaches: a case study in the gas and oil industry. *20th International Conference of the System Dynamics Society*, Jul. 28-Aug. 1, Palermo, Italy.
- Jones, A., Y. Yih, and E. Wallace. 2001. Mentoring and controlling operations. Chapter 65 in *The Handbook of Industrial Engineering: technology and operations management*, ed. Gavriel Salvendy. New York, NY:Wiley.
- Law, A., and W. Kelton. 2000. *Simulation modeling and analysis*. McGraw Hill, USA.
- Lee, Y., M. Cho, S. Kim, and Y. Kim. 2002. Supply chain simulation with discrete-continuous combined modeling. *Computer and Industrial Engineering* 43:375-392.
- Lin, C., T. Baines, J. O'Kane, and D. Link. 1998. A generic methodology that aids the application of system dynamics to manufacturing system modeling. *International Conference on Simulation*, Sep 30 – Oct 2 (IEEE Conf. Pub. No. 457).
- Mandal, P., and A. Sohal. 1998. Modeling helps in understanding policy alternatives: A case. *Journal of Management in Engineering* Jan-Feb: 41-48.

- Pegden, C., R. Shannon, and R. Sadowski. 1990. *Introduction to simulation using SIMAN*. McGraw-Hill, USA.
- Pfeiffer, A., B. Kadar, L. Monostori, and D. Karnok. 2008. Simulation as one of the core technologies for digital enterprises: assessment of hybrid scheduling methods. *International Journal of Computer Integrated Manufacturing* 21(2):206-214.
- Rabelo, L., M. Helal, A. Jones, and H. Min. 2005. Enterprise simulation: A hybrid system approach. *International Journal of Computer Integrated Manufacturing* 18(6):498-508.
- Rockwell Automation 2010. Arena Simulation Software. Available via <http://www.arenasimulation.com/> [accessed April 14, 2010].
- Sinreich, D., and B. Shints. 2006. A robust FMS control architecture with an embedded adaptive scheduling mechanism. *Journal of Manufacturing Systems* 24(4):301-312.
- Smith, J. 2003. Survey on the use of simulation for manufacturing system design and operation. *Journal of Manufacturing Systems* 22(2): 157-171.
- Son, Y., and J. Venkateswarana. 2005. Hybrid system dynamics discrete event simulation based architecture for hierarchical production planning. *International Journal Production Research* 43(20):4397-4429.
- Sterman, J. 1989. Modeling managerial behavior: misperception of feedback in a dynamic decision making experiment. *Management Science* 35(3):321-339.
- Sterman, J. 2000. *Business dynamics: systems thinking and modeling for a complex world*. New York, USA: McGraw Hill.
- Towill, D., and J. Edghill. 1989. The use of system dynamics in manufacturing systems engineering. *Transactions of the Institute of Measurement and Control* 11:208-216.
- Ventana Systems 2010. Vensim. Available via <http://vensim.com/> [accessed April 14, 2010].
- Zulch, G., U. Jonsson, and Fischer J. 2002. Hierarchical simulation of complex production systems by coupling of models. *International Journal of Production Economics* 77:39-51.

AUTHOR BIOGRAPHIES

JOHN PASTRANA received his Masters in Engineering Management from the University of Central Florida. He is currently pursuing a Ph.D. in Industrial Engineering at the University of Central Florida. His research interests are mainly in simulation and modeling of business enterprises, supply chain management and operation management. His email is pastranaja@knights.ucf.edu.

MARIO MARIN received his Masters in Simulation Modeling and Analysis from the University of Central Florida. He is currently pursuing a Ph.D. in Industrial Engineering at the University of Central Florida. His research interests are mainly simulation integration of operations and HLA and XML technologies. His email is mmarin@knights.ucf.edu.

MAGDY HELAL graduated with the Doctoral Degree in the department of Industrial Engineering and Management Systems, University of Central Florida. He received a Masters degree in industrial engineering and a Bachelor in Manufacturing Engineering, from Benha Higher Institute of Technology (BHIT) in Egypt. Helal is researching in the areas of enterprise engineering, production planning and control, and hybrid continuous-discrete simulation modeling. His e-mail is helal_ie@yahoo.com.uk.

CARLOS MENDIZABAL is a Fulbright Scholar at the Technological University in Panama. He received dual degrees in Electrical and Mechanical Engineering from the Technological University of Panama and Master's degrees from the Florida Institute of Technology (Electrical Engineer 1987) and the University of Missouri-Rolla (Engineering Management, 1988). He received a Ph.D. in Engineering Management from the University of Missouri-Rolla in 1990 where he also did Post-Doctoral work in Nuclear Engineering. His e-mail address is luis_mit@yahoo.com.