The Integrating Role of Simulation in Modern Manufacturing Planning and Scheduling

Slavko Arsovski[1,*] - Zora Arsovski[2] - Zoran Mirovic[3]
[1] Faculty of Mechanical Engineering, Kragujevac, Serbia,
[2] Faculty of Economic Science, Kragujevac, Serbia,
[3] Center for Quality, Faculty of Mechanical Engineering, Kragujevac, Serbia

The basic management problems in today's manufacturing organizations are integration of strategic and tactical decision making, developing the capability for plans and schedules reconfiguration and synchronization in a very short cycle as well as handling supply, operational and demand exceptions. This is reason why many firms have proceeded to the adoption of modern technological solution such as Enterprise Resource Planning (ERP) and Advanced Planning and Scheduling (APS) technologies. The final goal is a purposeful scheduling system that guide in the right direction the current, high priority needs of the shop floor while remaining consistent with long-term production plans. Simulation's role in such a system is key due to its ability to faithfully replicate the real production ambiance and quickly react on unpredictable exceptions in the field. Its value is further enhanced when this solution is fully integrated with advanced planning function. It is apparent that there is a lack of a comprehensive, scientifically established design methodology to structure, guide, and improve manufacturing process modeling efforts. The approach presented in the paper gives verbal and mathematical problem description, builds ontology of problem domain, uses Extended Petri Nets and event graphs as activity cycle diagrams as modeling tools in order to obtain faithful model which can easily be replicated in object oriented class and object hierarchy. With proposed approach it is possible to focus, comprehensive but static ERP knowledge in dynamic simulation models and to fully utilize prediction power of simulation for effective and integrated strategic and tactical decision making.
Keywords: manufacturing, production planning, production scheduling

0 INTRODUCTION

The speeding up of supply chain pace in which customer orders are initiated and finished products are delivered is one of the basic problems manufacturers must compete with and conquer. The most universal challenge is in handling supply, operational and demand exceptions what leads to the apparently impossible mission of planning for the unplanned. The speed at which business system identify these exceptions and react in order to reduce its negative impact on overall system performance primarily depends on responsiveness of enterprise information system and its capability of handling the above three factors. The contemporary solution is integration of strategic and tactical decision making and, on production level, developing the capability for synchronization, modification and fine-tuning of production plans and schedules as quick as possible. Today, business systems try to achieve this goal through implementation of modern information technologies, specifically, through integration of ERP, APS and Discrete Event Simulation (DES) software systems [1].

Although an ideal ERP system should support the majority, if not all, of an enterprise's business processes, in the production area and, more specifically, in the production planning and scheduling field, there was a lack of satisfactory support. Since 2000 many well known vendors of ERP systems such as SAP, Oracle, Baan, PeopleSoft and others have included in its commercial solutions, up to this moment relatively independent APS systems with scheduling and certain, limited simulation capabilities and have recorded consistent growth in revenues [2]. Simulation's scheduling role in such a system is key due in large part to its ability to faithfully replicate the real production ambiance and quickly react on unpredictable exceptions in the field. Its value is further

---
*Corr. Author's Address: Faculty of Mechanical Engineering, 34000 Kragujevac, Serbia, cqm@kg.ac.yu

enhanced when this solution is fully integrated with advanced planning function.

Despite this fact, even now are evident problems in successful implementation and utilization of this scheduling, simulation supported solutions in everyday production practice. The main reason, what this paper suggests, is that in the domain of production scheduling it is not possible to design generic model (simulation software) capable to encompass all possible production system types and, in as much detail as it is necessary, all the subtleties of the concrete manufacturing environment. This is reason why above simulation solutions suffer the loss of its prediction power that is crucial for handling exceptions [1] and [4].

Simulation software systems are dynamic models of complex reality and this fact result in complexity of all stages of software creation. The paper also highlights unavoidability of knowledge transfer between three expert groups with different knowledge domains. First group are business processes experts, second software development experts and third from academic community research. Synthesis of mentioned fields of knowledge in proposed approach result in effective and functional simulation model.

The paper suggests methodological design approach for DES software solution. Proposed approach gives verbal and mathematical problem description, builds ontology of problem domain, uses Extended Petri Nets and event graphs as activity cycle diagrams as modeling tools, in order to obtain faithful model which easily can be replicated in object oriented class and object hierarchy [4] and [8].

1 ERP, APS AND DES IN TODAY'S BUSINESS

ERP systems are software solutions that provide seamless integration of all enterprise's business processes and information flows in the company attempting to synergize the resources of an organization. A typical ERP system is single software package that use multiple components of computer software and hardware to achieve mentioned integration. A distinctive constituents of most ERP systems is the use of a unified database, common data-processing and communications protocols to process and store data for the various system modules. ERP system combines all business applications together into a single, integrated software program that runs off a single database so that the various departments can more easily share information and communicate with each other.

Ideally, ERP delivers a single database that contains all data for the software modules or business application, which would include: manufacturing (engineering, bills of material, scheduling, capacity, workflow management, quality control, cost management, manufacturing process, manufacturing projects, manufacturing flow), supply chain management (inventory, order entry, purchasing, product configurator, supply chain planning, supplier scheduling, inspection of goods, claim processing, commission calculation), financials (general ledger, cash management, accounts payable, accounts receivable, fixed assets), projects (costing, billing, time and expense, activity management), human resources (human resources, payroll, training, time & attendance, benefits), customer resources and marketing (sales and marketing, commissions, service, customer contact and call center support) and various *self-service* interfaces for customers, suppliers, and employees

The largest vendors worldwide in 2005 according to Gartner Dataquest are: SAP (with revenue 4726 in million dollars and market share 28.7%), Oracle Applications (1674, 10.2%, The Sage Group (1221, 7.4%), Microsoft Dynamics (616, 3.7%), SSA Global Technologies (464, 2.8%).

Planning and scheduling are closely related and are best done using the same or closely related tools like APS. Scheduling determines what is actually implemented in the process, while planning explores what is possible. Plans have little value if they are inaccurate and cannot be implemented because they do not consider important process requirements [2] and [4]. As such a contemporary APS system should support both planning and scheduling. With this in mind an APS system can be defined as computer software with ability to rapidly and simultaneously plan and schedule customer demand while considering material and capacity constraints as well as to reconfigure synchronized plans and schedules in a very short cycle time. The potential to increase business performance

such as on-time delivery, shorter cycle times, reduction in inventories and increased throughput with APS is enormous.

The available data suggests that the estimated improvement attributable to effective planning and scheduling is 5 to 15% as measured by a decrease in process costs (such as waste, changeover, inventory reduction) and/or increase in process throughput. In order to achieve this performance an APS system must be used in an effective business process whereby the data used is reasonably accurate and schedules and plans must be executed with reasonable precision. To achieve consistent results the planning and scheduling process must be repeated when business conditions change significantly.

Low end APS systems cost on the order of several hundred or a thousand US dollars. Mid-range APS systems cost several tens of thousands of US dollars to install. The most expensive APS systems generally cost about $500,000 to install, where half to two thirds of the cost is consulting fees to customize the system to a particular process. The purpose of an ERP is to provide data to those needing it, therefore technical integration of APS systems with ERP data system is usually straightforward.

2 PLANNING AND SCHEDULING IN ERP/APS AMBIANCE

The computerized process of planning and scheduling in manufacturing have evolved from simplistic Material Requirements Planning systems to today's sophisticated APS systems. While planning is concerned with the long-range determination of what needs to be manufactured, typically over a relatively long time period, scheduling is the task of deciding how that manufacturing is to be accomplished, usually over a relatively short time period. Simulation is well suited to the scheduling task since it can handle as much detail as is necessary to capture the subtleties of the manufacturing process. It is desirable for a simulation-based scheduling function to be integrated with an ERP system, which maintains the system data suitable for driving a simulation of the current system load and thereby producing a feasible schedule [2] and [8].

Modern ERP systems contain all the data necessary for detailed production planning and scheduling. This includes product information, such as bills of material and routing of parts through the manufacturing process. It includes system information such as equipment, manpower, and shift schedules. It also includes status in formation such as the current order book, work in process, inventory levels, and released purchase orders. This is what is needed for an APS function to determine how to efficiently plan a plant's operations and to replan quickly and accurately based on changing requirements [1] and [3].

The traditional Material Requirements Planning (MRP) computerized process in an ERP system concerns itself with determining the quantity of products to be made in a specified period of time. Initial information are demand for final products in that period, the component products which compose that demand, and the lead time required to produce each component and final product. A major defect in such an approach, of course, is that limits on manufacturing capacity are not considered [1] and [3]. Actual lead times usually vary considerably from the fixed lead times assumed by MRP when a system is highly utilized and dynamic. In order to determine whether or not an infinite capacity plan such as produced by MRP is actually feasible, simulation can be used to determine whether the start times generated by the plan will actually allow the manufacturing orders to be completed by their due dates. The integration of simulation with manufacturing planning systems became feasible in the 1980's as computing power increased to support production control systems, shop floor data collection systems and database management systems.

While simulation is capable of producing a highly realistic manufacturing schedule, the task of "correcting" the infeasibilities of an MRP plan in capacity-constrained environments is quite discouraging. Through the years the authors have been involved in many successful applications of such, but the data requirements to maintain a realistic model consistent with the plan and the business process expertise needed to effectively execute it have made these successes costly and difficult to perpetuate. What is needed is a better starting point from which to schedule - which leads to APS systems.
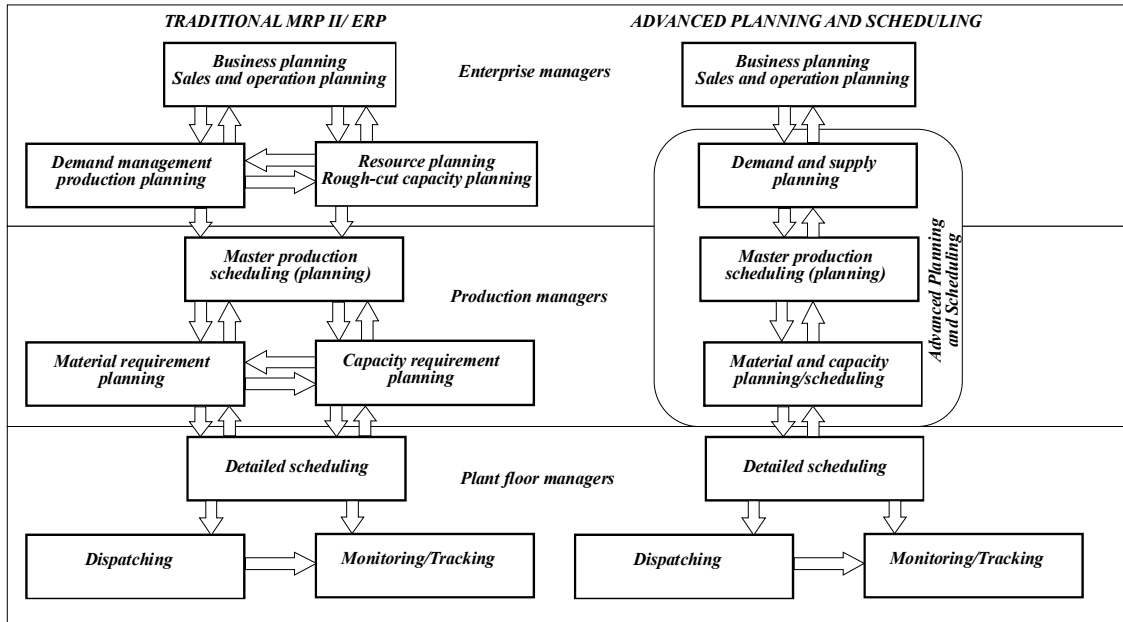
Fig. 1. *Traditional and advanced planning and scheduling system*

An APS system uses variants of the planning and scheduling approaches described above in an integrated way. A planner module which pays some attention to capacity constraints produces a "schedulable" plan. This plan then feeds a scheduler module, which produces a detailed list of operations showing how capacity will be used and returns this information to the planning function for use in the next planning period. The data regarding current and planned operations can also be used to provide realistic estimates of the ability to meet a new customer order request. Function of an APS system is to coordinate material and capacity planning in order to fulfill the demands being placed on the manufacturing system. It does this using three key processes: advanced planning, advanced scheduling, and order promising.

The role of planning in APS is to determine what demands on the production system will be met over a given planning horizon. The input to the planning process includes information on manufacturing capacity and demand data. Demands may be of several types: customer orders, forecasts, transfer orders (i.e., orders from other plants), released jobs, or replenishments of safety stock. Manufacturing system data includes bills of material, work center availability, part routings through work centers, and inventory (both on-hand and scheduled for

delivery). The output from the planning process is a feasible plan, which provides release and completion times for every demand. Like MRP before it, APS takes into account the availability of materials. Unlike MRP, it also takes into account the capacity of work centers to process the material and satisfy demands. This planning process is order-centric, focusing on the demand for end items and determining how much demand can be met in a given time period. Exactly how that demand will be met, in terms of specific assignments of jobs to work centers and their sequencing, is left to the scheduling function. It is in fact often desirable for a plan to be somewhat tentative, since it covers a planning horizon subject to disruptions. Forecasts may not be accurate. Deliveries may be delayed. Equipment may fail. Unexpected rush orders may be received. Therefore planning is not expected to be highly detailed. Individual machines may be aggregated into a work center with no determination of which will be used by a specific order. Setup times may be averaged since sequencing at this time is premature. Buffer times may be defined, especially prior to processing on bottleneck machines, to allow for possible disruptions. The end result is a "schedulable" plan.

The logic of the planner algorithm operates on each order in turn, as follows [1]:

- Start backward from the promise date and reserve the work center capacity and material required for each operation. If the material required is a manufactured component, then it too is planned backward from the point at which it is required. If at the end of this pass the start date is on or after the beginning of the planning horizon, proceed to the next order.

- If the backward pass fails, meaning that the resulting start date would be in the past, reserve capacity and material forward from the planning start date. If the resulting completion date is within the planning horizon (even though after the promise date) this order is planned.

If both the backward and forward passes fail, then the order is not included in the planning period. The end result of the above process is a "feasible" plan.

The role of the scheduler module in APS is to produce a detailed list of operations specifying which orders are to be worked on at which work centers and at what times. The input to this module includes all demands to be satisfied, including the internal orders added by the planner module when an end item required a component to be manufactured. It includes the current material inventory levels as well as planned deliveries of purchased materials. It also includes the same manufacturing system data as that provided to the planner module but uses a more detailed representation of that data. Detailed information used by the scheduler module that is not pertinent to the planner module includes:

- Variable run times based on the machine and operator actually assigned

- Rules for selecting machines and operators based on skill sets and quality requirements

- Variable setup times based on the previous and next part characteristics such as part type, family, color, width, etc.

- Rules for sequencing jobs at work centers, based on minimizing setup and other factors

- Acceptable shift overruns.

- Rules for selecting from a list of prioritized jobs based on due date, slack, cost and other factors.

The result is an accurate representation of what to expect on the shop floor in the immediate future. While the planner module typically considers demand on the system over a few weeks or months, the scheduler module will typically work with a much shorter time frame such as a shift, a day, or a week. The usefulness of a detailed schedule degenerates quickly as time passes, since disruptions on the shop floor or changes to the order mix may require significant adjustments. For this reason a simulation used for generating a schedule is usually deterministic. If a random event occurs (i.e., machine failure, arrival of a rush order, or a missed delivery date by a supplier) then a new schedule can quickly be generated and its impact evaluated.

2.1 Scheduling Using Simulation

While the APS planning function is order-centric, the APS scheduling function is event-driven. Given a set of orders and associated start dates, the scheduler algorithm begins by generating a calendar, that is, a time-ordered list containing the first scheduled operation for each order. As each operation is able to acquire the work center capacity and materials it requires, the calendar is updated to reflect the time at which the operation will end. Each order moves through its route over time, with its completion date dependent on the dynamics of the system. Which specific machine will it be allocated at a given operation? How long will it have to wait for other orders at the same machine? What will the setup time be? Answers to these questions cannot be planned ahead of time, but rather unfold dynamically as orders move through the system over time. The end result is a detailed schedule. Fig. 2 illustrates that the scheduled jobs dictated by the scheduler module and purchase orders released by the ERP system are returned to the planner module for the next planning run. It is this put back that allows the APS Planner to continue to produce realistic plans while honoring the APS schedules. The entire planning and scheduling process is tightly integrated, but not automatic. As described earlier, the APS Planner produces planned manufacturing orders and planned purchase orders.
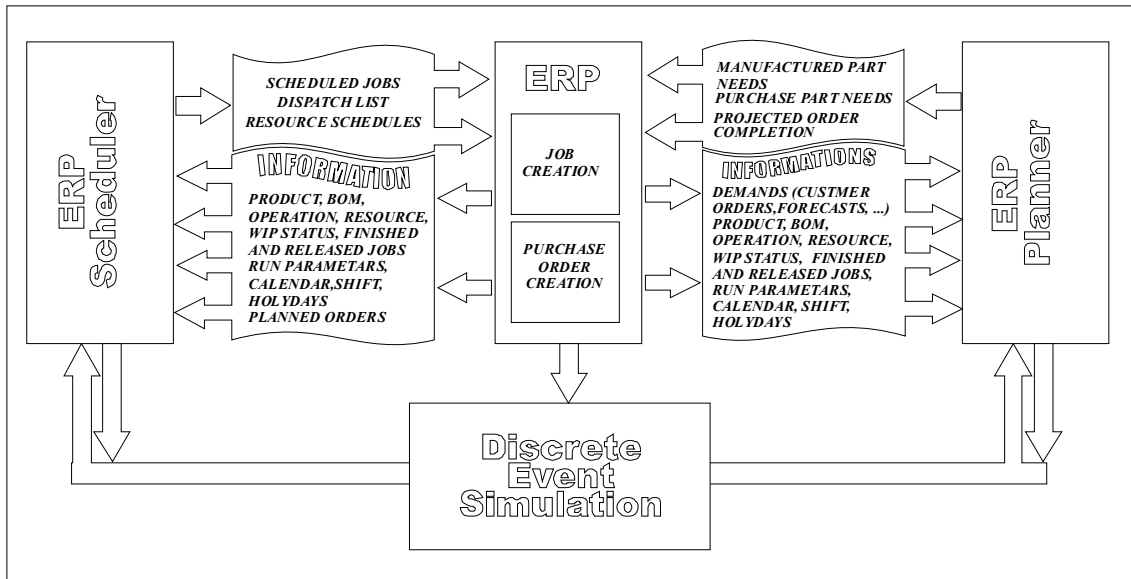
Fig. 2. *Closing the planner/scheduler loop with simulation*

These are based on actual customer orders and, optionally, on forecasts and other estimates. A (human) production planner now may use these recommendations to release jobs to the shop floor and to create purchase orders. The APS Scheduler is then run, and uses these planned and released jobs to assign work center capacity and inventory. Only the released job assignments are then fixed within the APS Planner to create new recommended plans.

A feasible schedule may not be a desirable schedule. If too many orders are not "schedulable", for example, there are a number of things that might be tried to temporarily in crease the capacity of the system. Perhaps an extra shift could be added for a key work center. Temporary help might be brought in if labor is the bottleneck. One or more operations might be outsourced. A major benefit of simulation has always been the ability to investigate variations on a system without disturbing its operation. This benefit has been brought to APS by providing the ability to copy the ERP data and experiment with changing one or more parameters and comparing the results with the original.

The ability of an APS system to realistically project order completion times as well as the utilization of capacity and materials is critical in today's competitive manufacturing world. Combining APS planning technology with enterprise management processes can improve customer service and on-time delivery, reduce expediting, overtime and inventory, and increase throughput and profit.

3 PROPOSED DESIGN METHODOLOGY AND REVIEW OF CONCRETE SOLUTIONS

Simulation software systems attempt to model or replicate complexity of real systems what result in complexity at analytical, design and technological level of software creation. This is reason why effective modeling is one of the most important and difficult steps in the development of reliable simulation software systems. At this point, the paper highlights inevitability of knowledge transfer between three knowledge domains. First domain is knowledge about the business processes, second knowledge about software development and implementation and third from academic community research (Operational Research and Management Science) about problem domain. Integration and overlapping of mentioned specialized fields of knowledge for the solution of the common task result in useful and functional simulation model. In this framework the paper proposes methodological design approach for Discrete Event Simulation solution (Fig. 3.). This approach gives verbal and mathematical problem description, builds ontology of problem domain, uses Extended Petri Nets and event graphs as activity

cycle diagrams as modeling tools, in order to obtain faithful model which can easily be replicated in object oriented class and object hierarchy.
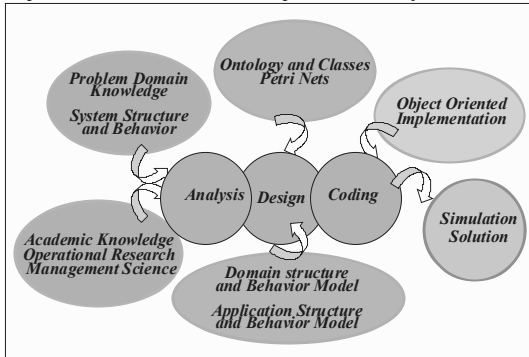

Fig. 3. *Proposed methodology*

3.1 Basic Concepts

Production scheduling is defined as a process of realistically synchronizing the use of resources by activities to satisfy orders over time. An order is an input request for one or more products, which designate the goods or services required. Satisfaction of orders centers around the execution of activities. An activity is a process that uses resources to produce goods or provide services. The use of resources and the implementation of activities are restricted by a set of constraints. These five basic ontology – order, activity, resource, product and constraint – together with the inter-relationships define an abstract model of a production scheduling domain. Of course, there are many other ontology necessary for the successful design of a production scheduling information system [5]. In the following subsections, we analyze the basic ontology of the methodology in more detail.

Order:

- *Concept definition.* An order is a specific request for goods and/or services, or more generally products, which the system being modeled can produce. Orders specify the input goals that drive the system, along with any constraints that should be taken into consideration when achieving them.
- *Properties.* An order has several defining properties: Order_code, Product_code, Quantity, Due_date, Priority, Order_status.

Product:

- *Concept definition.* A product is a good realized through the execution of some set of activities. An order for a product is considered satisfied when all of these activities have been completed.
- *Properties.* The product definition includes the following: Product_code, Routing_code.

Routing:

- *Concept definition.* The routing indicates the specific set of activities which must be implemented so that a product is produced, as well as their sequence.
- *Properties.* The routing definition includes the following: Routing_code, Activity_code. Activity_range.

Activity

- *Concept definition.* An activity represents a process that can be executed over a certain tine interval. An activity requires resources to be executed and its execution both depends on and affects the current state of these resources.
- *Properties.* The basic decision variables associated with an activity are: Activity_code, Product_code, Start_time, End_time, Workcenter_code, Activity_range, Activity_status.

Resources**:**

- *Concept definition.* Central to the definition of our scheduling ontology is the concept of a resource. A resource is an entity that supports or enables the execution of activities. Resources are generally in finite supply and their availability constrains when and how activities can be implemented. In a manufacturing system, a resource can be a machine (or a work center in general) and/or human resource needed for the proper execution of an activity.

In general, we can distinguish two broad classes of resources from the standpoint of availability:

- Capacitated-resources, whose availability is characterized in terms of the amount of capacity that is available,
- Discrete-state-resources, whose availability is a function of some discrete set of possible state values.

It is stressed that the resources, together with their properties (machine_capacity, machine_code, technician_code, etc.) are embodied in the other basic ontologies.

Work center:

- *Concept definition.* A work center is an area in a business in which productive resources are organized to implement a specific work. A work center may be a single machine or a group of similar machines. These work centers can be organized according to functions in a job-shop configuration, or by product in a flow-shop (assembly line).
- *Properties.* The work center definition includes the following: Workcenter_code, Activity_code, Average_setup_time, Average_capacity.

Machine:

- *Concept definition.* A machine is considered to execute only one activity every time and it is specified by the scheduling system.
- *Properties.* The properties of a machine are the following: Machine_code, Workcenter_code, Machine_capacity, Machine_availability, Machine_flag.

Personnel:

- *Concept definition.* An activity can be executed by using machines and human resources (personnel) as well. The human resources concern both the necessary setup of a machine and the whole execution of an activity.
- *Properties.* The properties of personnel are the following: Technician_code, Product_code, Activity_code, Machine_code, Setup_input, Activity_input.

Constraints:

- *Concept definition.* Generally speaking, a constraint restricts the set of values that can be assigned to a variable. In the production scheduling domain, constraints restrict the assignment of start and end-times and the allocation of resources to activities.
- *Properties.* From this perspective, we can identify two basic types:
  - Hard. The production scheduling system is never allowed to violate hard-constraints.
  - Soft. Alternatively, soft-constraints are considered to be relaxable, if needed so. For example, due-date constraints are treated as relaxable constraints in many scheduling systems.

Possible restrictions and constraints which may appear in a production scheduling system are the following:

- Release dates. A specific "job" may not start its processing before its release date.
- Preemptions. Preemptions imply that it is not necessary to keep a "job" on a machine until completion. The scheduler may interrupt the processing of a "job" (preempt) at any time and put a different "job" on the machine. The amount of processing a preempted "job" has already received is not lost. When a preempted "job" is put back on the machine (or on another machine), it only needs the machine for its remaining processing time. Set up time is defined in work center properties.
- Precedence constraints. Precedence constraints may appear in single machine or in parallel machine environments, requiring that one or more "jobs" may have to be completed before another "job" is allowed to start its processing.
- Permutation. A constraint that may appear in the flow-shop environment is that the queues in front of each machine operate according to the FIFO discipline. This implies that the order in which the "jobs" go through the first machine is maintained throughout the system.
- Blocking. Blocking is a phenomenon that may occur in flow-shops. If a flow-shop has a limited buffer in between two successive machines, the upstream machine is not allowed to release a completed "job".
- Recirculation. Recirculation may occur in job-shops, when a job may visit a machine more than once.

3.2 Mathematical Specification

A scheduling task can be formally represented as a mapping from a nine-dimensional space {J, A, R, C, Req, Tr, P, Cf, Cr} to a set of schedules S = {S_1,...., S_n}. These parameters are described as follows:

- *Job,* J = {j_1, ..., j_m}. A set of jobs to be assigned to a set of resources for their execution.
- *Activities,* A = {a_1, ..., a_n}. A set of activities. Each activity must be associated to a job, and no activity can be associated to more than one job.

- *Resources,* R = {r_1, …., r_i}. A set of resources to which the jobs can be assigned for their execution.
- *Constraints,* C = {c_1, …., c_j}. A set of constraints that must not be violated by a solution schedule.
- *Requirements,* Req = {req_1, …., req_k}. A set of requirements that describe the desired properties of a solution schedule.
- *Schedule time range,* Tr. The time horizon in which the schedule takes place.
- *Preferences,* P = {p_1, …., p_t}. A set of criteria for choosing among competing solution schedules. Each preference defines a partial order over the set of solution schedules.
- *Cost function,* Cf. A function, which computes the cost of a final solution schedule.
- *Solution criterion,* Cr. A mapping from S to {True, False}, which determines whether a candidate schedule is a solution. The minimum set of conditions imposed by a solution criterion on a solution schedule, say S_i, normally requires S_i to be *correct, complete, admissible,* and *feasible* - see below for the definitions of these properties. More restrictive solution criteria may introduce optimality conditions based on the applicable preferences and cost function.
- *Schedule,* S = {…<j_i, a_{ij}, r_k, tr_{ijk}>…}. A schedule is a set of quadruples <j_i, a_{ij}, r_k, tr_{ijk}>, where j_i is a job, a_{ij} is an activity associated with j_i, r_k is a resource, and tr_{ijk} is the *job time range* assigned to activity a_{ij} of j_i. A job time range specifies the earliest and latest start time and the duration of a_{ij}. We assume that no additional control information is required to interpret a schedule. In other words, we assume that the execution of a schedule only requires assigning resources to jobs and activities, in accordance with the given time ranges.
- Criteria to check the validity of a schedule.
- A schedule, say S_i, is *correct,* if no pair <j_i, a_{ij}> appears in more than one quadruple in S_i.
- A schedule, say S_i, is *complete,* if for each activity a_{ij} in A, associated with job j_i, there exists a quadruple q in S_i, such that q = <j_i, a_{ij}, r_k, tr_{ijk} >.

- A schedule, say S_i, is *admissible,* if it does not violate any of the applicable constraints.
- A schedule, say S_i, is *feasible,* if it satisfies all requirements in Req.
- A solution schedule, say S_{opt}, is *optimum,* if there is no other solution schedule that has a lower cost than that of S_{opt}.

This approach uses Extended Petri Nets (EPN) like graphical and mathematical modeling tools that can be used to perform static and dynamic modeling of existing manufacturing systems. Manufacturing systems are characterized as being concurrent, asynchronous, distributed, parallel, non-deterministic, and stochastic and can be effectively modeled and analyzed by using Petri Nets.
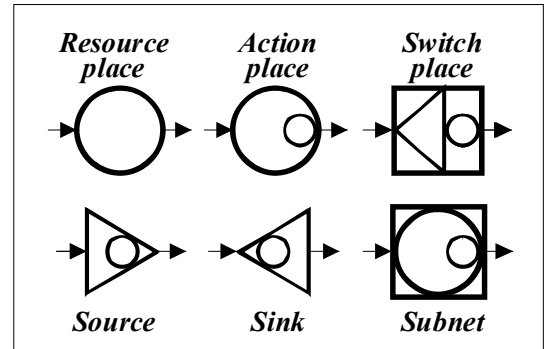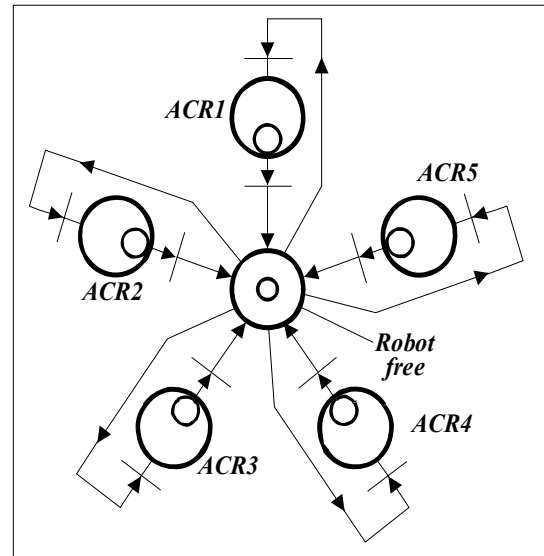

Fig. 4. *EPN symbols*

Fig. 5. *Activity cycle diagrams*

A place - denoted by a circle - represents a condition such as input data, input signal,

resource, condition, or buffer. A transition - denoted by a solid bar - represents an event such as a computation step, task, or activity. Arcs are utilized to connect places and transitions in a Petri Net. Arcs are directed (depicted by arrows) and are either drawn from a place to a transition or from a transition to a place. Arcs in a Petri Net have multiplicity. The fourth element called the token and denoted by a solid circle provides the dynamic simulation capabilities to Petri Nets. Tokens are initialized at a place and a place may contain zero or more tokens. With the use of tokens the modeler can provide the necessary dynamic links between the places (conditions) and transitions (tasks or events) in a Petri Net. The concept of transition "firing" allows a Petri Net to simulate the dynamic behavior of a system. In their original form transition firing in Petri Nets was instantaneous, but, time is incorporated into Extended Petri Nets. This results in a timed transition that will have the ability to model tasks or activities.

3.3 Review of Concrete Solutions

Ontology concepts are the base for object oriented class and object hierarchy design and EPN approach helps modeling of manufacturing system and software system dynamics. Realized simulation tool has a number of unique characteristics such as interactive Gantt chart display, specialized reports, integration with external data sources, specialized scheduling rules, concurrent graphical animation etc.

The quality of the generated schedule is largely determined by the scheduling rules that are specified for selecting resources and operations. A complete set of rules must be incorporated into the simulation tool to support a specific range of given manufacturing identity.

Figures 6 to 8 presents an EPN Algorithm of simulation model execution and screens of simulation solutions developed for different manufacturing environments. Figure 7 depicts a screen with concurrent graphical animation of flexible manufacturing system shop floor with four machines. Simulation software enables visual presentations to the user of all system events in corresponding time interval. Figure 8 presents Gantt diagram depicting complete system behavior with machine utilization and maximum flow time of job orders realization.

Implementation of realized software solutions can accelerate the flow of the right information, quality and speed of decision making what results in increasing of overall system performance.
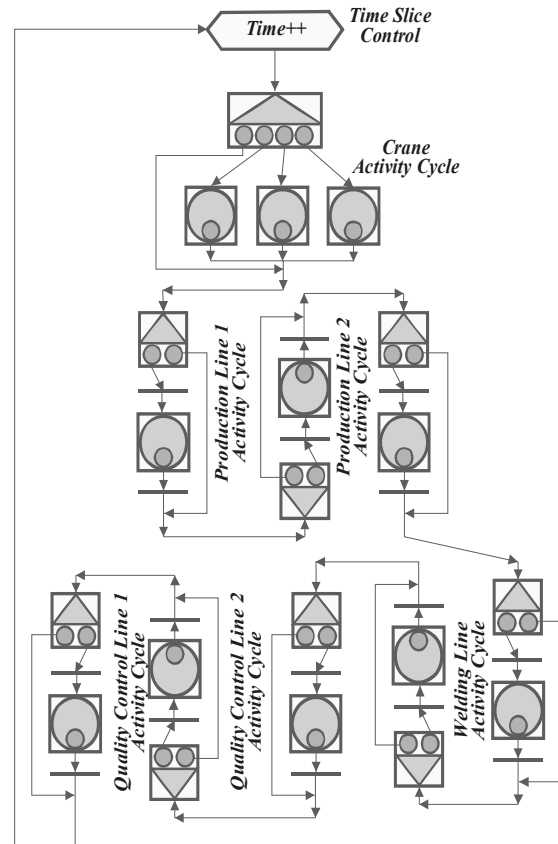

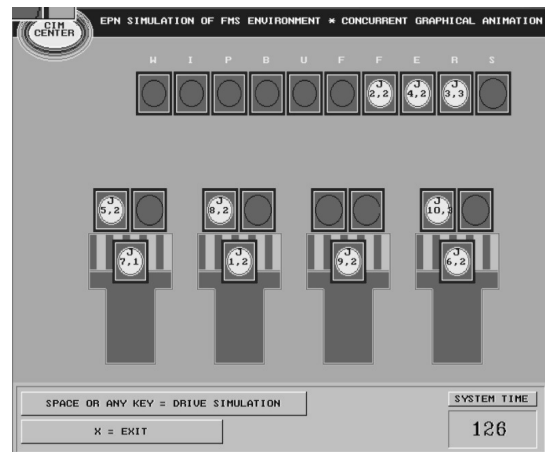Fig. 6. *EPN Algorithm of simulation model execution*

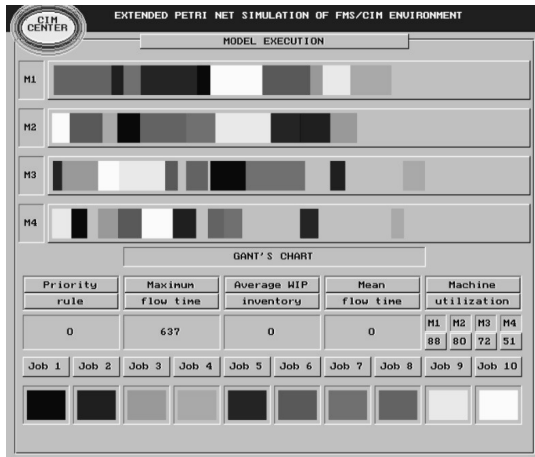Fig. 7. *Concurrent graphical animation*

Fig. 8. *Gantt chart*

4 CONCLUSIONS

One of the basic management problems today is how to manage a business process and at cost effective manner adapt to unavoidable change dictated by the environment. This causes problem in designing corresponding simulation software models which will replicate complexity, enable evolution control of modeled system as well as synchronization of its own growth with named evolution? The simulation model designed by methodology proposed in the paper tries to give an answer to this challenge. This approach gives verbal and mathematical problem description, builds ontology of problem domain, uses Extended Petri Nets and event graphs as activity cycle diagrams as modeling tools in order to obtain faithful model which easily can be replicated in object oriented class and object hierarchy. This approach also gives emphasis to unavoidability of knowledge transfer between business processes experts, software development experts and academic community research groups. With proposed methodology many kinds of integration with an ERP/APS system will be dynamically available for floor level management through dynamic simulation models. This approach facilitates integration of strategic and tactical decision making, developing the capability for plans and schedules reconfiguration and synchronization in a very short cycle as well as handling supply, operational and demand exceptions.

A dynamic simulation-based approach to production planning and scheduling has distinct advantages over commonly used static approaches. The most vital of these is the direct and dynamic link to the shop floor. Simulation has the ability to accurately model detailed system operating rules and evaluate them over time what is invaluable in ensuring that a plan or schedule is feasible on the shop floor level. Sudden changes in employee availability, machine breakdown, extra orders, order expediting requirements, etc. can be reacted quickly using simulation to generate a new and efficient schedule within minutes. It is difficult to assess the feasibility of schedules generated by static methods other than by actually executing it in real time on the shop floor. This limits the opportunity to anticipate inefficient or unfeasible plans in advance. Solutions can be fully customized according to the current planning and scheduling rules and incorporated in modern ERP/APS environment. Integration of planning and scheduling with simulation bridge the gap between tactical and strategic management information systems result in a more coordinated production planning and execution.

5 REFERENCES

[1] Musselman K.J., Uzsoy R. Advanced planning and scheduling for manufacturing. In G. Salvendy, *Handbook of Industrial Engineering* (3rd ed.), 2001, New York.

[2] Metaxiotis K.S., Psarras J.E, Ergazakis K.A. Production scheduling in ERP systems: An AI-based approach to face the gap, *Business Process Management Journal,* vol. 9, no. 2 2003 p. 221-247.

[3] Metaxiotis K. S. Psarras J. E., Askounis D. Building ontologies for production scheduling systems: towards a unified methodology**, *Information Management & Computer Security,* vol. 9, no. 1, 2001, p. 44-51.

[4] Anglani A., Grieco A., Pacella M., Tolio T. Object oriented modeling and simulation of flexible manufacturing systems: A rule-based procedure, *Simulation Modelling Practice and Theory 10*, 2002, p. 209-234.

[5] Smith S. F., Becker M. A. An Ontology for Constructing Scheduling Systems. *Proceeding of AAAI-97, Spring Symposium on Ontological Engineering*, 1997.

[6] Mirovic Z. Quality through Integration of Production and Shop Floor Management by Discrete Event Simulation, *2nd International Quality Conference, Kragujevac*, 2007.

[7] Arsovski Z., Arsovski S., Mirovic Z**.** Integration of Just In Time and MPII

concepts by simulation software, *INFOFEST ′98*, Budva, 1998.

[8] Park H., Kim Y., Park S. An object oriented production planning system development in an ERP environment, *Computers in Industrial Engineering,* vol. 35, no. 1-2, 1999, p.157-60.

Arsovski, S. - Arsovski, Z - Mirović, Z.