Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds

FORMAL MODELS FOR ALTERNATIVE REPRESENTATIONS OF MANUFACTURING SYSTEMS OF SYSTEMS

Seungyub Lee

Department of Industrial and Manufacturing Engineering
Pennsylvania State University
University Park, PA 16802 USA

Richard A. Wysk

Department of Industrial and Systems Engineering
North Carolina State University
Raleigh, NC 27695 USA

Dongmin Shin

Department of Industrial and Management Engineering
Hanyang University, ERICA Campus
Ansan, KOREA

ABSTRACT

Two separate approaches have been pursued to model manufacturing systems: a periodic process-oriented planning view and a discrete event-based operational view. It is desired to integrate both approaches. To meet this requirement, this paper presents formal descriptive models for a manufacturing supply chain system which can be assembled to unite heterogeneous system views. These models can be used to coordinate complex hierarchical manufacturing systems. The formal description of a system model consists of: (1) a Discrete Event System (DES)-based operational model of the physical system processes for system flows, (2) a periodic review-based planning model for decision-making processes for system coordination, and (3) an interaction and a temporal model for enabling the communication between the two above models. The model presented in this paper can be used to implement more realistic and seamless manufacturing system control mechanisms with consideration of logical planning and physical operational aspects at the same time.

1 INTRODUCTION

For large and complex Systems of Systems (SoS), autonomous and distributed sub-systems should be properly represented using suitable modeling methodologies. As the size and complexity of problem domains increases, a variety of system aspects need to be considered in a way that models can be extended in accordance with the problem size and accommodate various viewpoints of a problem with respect to how to architect, model, and execute software applications. Accordingly, two key challenging questions have emerged: (1) How can proper definitions and specifications be developed for the target problems to provide consistent software architectural synthesis among different participants? and (2) How can efficient and seamless integration and coordination mechanisms be developed in the descriptive model?

In order to represent physical and logical manufacturing systems and an appropriate coordination framework, we provide a set of formal models to describe system entities, their relationships, behaviors, interactions, and coordination rules in multiple abstract levels. This formal model serves as a linkage between different abstractions of manufacturing system models, i.e., a federation of ERP (transactional planning decision) systems and multi-level manufacturing (operational) simulation models. As recent manufacturing systems increasingly require subtle coordination of sub-systems with different abstract levels, a systematic formal model for accommodating different modeling views becomes more important.

To illustrate the formal model presented in this paper, control of a manufacturing supply chain system is considered. The coordination of a SoS is also addressed in a supply chain context. Simulation is employed to model a DES-based operational view of supply chain components. In order to consider a periodic planning aspect in coordinating the federation, Supply Chain Operation Reference (SCOR) and an Enterprise Resource Planning (ERP) system are used to represent planning components.

Structured formal methods are appropriate for use in software development and offer broad notations that can be understood by non-specialists or third party sub-contractors. Structural software development has produced notable research results (Zeigler, Praehofer, and Kim 2000; Smith, Joshi, and Qiu 2003; Shin, Wysk, and Rothrock 2006) for scalable system modularity and component coupling for modeling and reusable simulation layers, that can be used for the execution portion of shop floor control. In spite of significant advances in formal models of simulation, it is still in need to develop a model that can take process-oriented and DES-based views into account at the same time.

To meet this need, this research 1) structures system parallelism for both transactional (planning model) and DES-based (operational model) approaches, 2) produces a set of notations for the semantic library for a SoS, and 3) defines a set of operators which can be directly compiled into a coordination system. The descriptive models discussed in this paper provide basis for a hierarchical coordination framework for a distributed software federation consisting of a set of simulation models and transaction managers. The paper is organized as follows. A set of key functional requirements for system configuration and the scope of the manufacturing supply chain simulation are introduced in section 2. In section 3, two different system viewpoints related to process-oriented and event-based approaches are discussed. Section 4 presents the structural modeling formalism for the system components and the behavioral modeling formalism consisting of planning, operational, temporal and interaction models. Lastly, research summary and possible future direction of the research are discussed in section 5.

2 THE FUNCTIONAL REQUIREMENTS AND SCOPE FOR THE CONFIGURATION OF PHYSICAL AND VIRTUAL SYSTEMS

The fundamental issue in developing a manufacturing supply chain system is to identify key elements and their complex relationships in the functional areas such as a marketing/sales department, planning functions, and a manufacturing shop floor. In a supply chain system, several different system entities must be considered such as spatial distributed organizations, hierarchical processes, and multiple model layer dealing with heterogeneous system objects with consideration of the structure of parts and processes.

The system entities have two very important characteristics: states and behavior. For example, parts can take states of "packaged@MP" or "assembled@MP" as a result of a process at a resource of a Material Processor (MP). Likewise, a few examples regarding states of transactions or orders can be "order received from SD", "order denied to SD" as a result of behavior at functional unit of a sales department (SD). Behaviors associated with parts can be "delivering", "ordering", or "manufacturing". "Quote receiving", or "sending receipt" can be behaviors related with orders.

Like the modeling approach used for the Message-based Part State Graph (MPSG) (Shin, Wysk, and Rothrock 2006; Smith, Joshi, and Qiu 2003; Son et al. 2002), the physical system in this research also adopts a part focused view which models the transitions of the material movements in a supply chain environment using hierarchical Mealy machine. Based on the aggregation level of the system objects, the graph can be represented as various viewpoints of the system entities and resources (either physical or virtual), e.g., a part, a batch, an order as shown in Figure 1. For example, a part-state graph of a part is usually constructed at the lowest level of the resource granularity which is associated with a physical machine or a material handler. In addition, a batch-state graph of parts can be constructed at the next high level of the resource granularity, e.g., organization's functional units.

A state-graph is composed of a set of vertices and a set of edges such that *G* = <*V, E*>. A vertex $v_k \in V$, denoted by an integer, represents a part or batch position in the graph and an edge $e_j \in E$ is a resource's physical operation associated with the part or batch. The vertices in a graph can be aggregated into a small number of vertices in which case an edge connecting aggregated vertices represents an interaction between functional units within an organization such as a procurement department, a manufacturing facility, a sales department, etc.

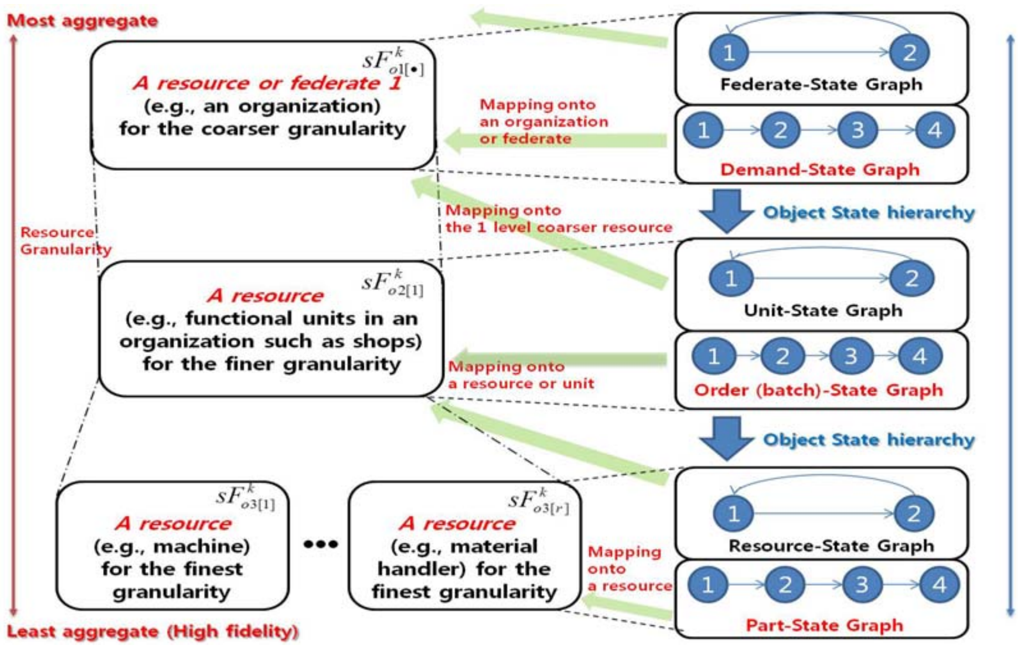


Figure 1: Relationship between the resource granularities and object state hierarchy for different state viewpoints and aggregations

The logical planning system for transaction managers such as Material Requirements Planning (MRP) or Master Production scheduling (MPS) is also represented by a graph-based view. States of a planning unit are determined by results of a business process within or between functional units concerning transactions or orders which are described by edges. For example, it decides how many parts should be released into a shop and whether the payment for an order is confirmed. While the part-state execution model controls operations in a single resource, a logical model such as Enterprise Resource Planning (ERP) system is the decision maker for an organization or task generation for physical operation systems. It governs states of functional units by allocating and coordinating tasks into each functional unit as shown in Figure 1.

Because of the interdependences among physical manufacturing operations and logical planning systems which can be different in abstract levels and time progress, and output of a system can be input to the other, it is necessary for the model to include: 1) time management for the simulation federation, 2) data management for the transaction manager federation, 3) an extended structural set for the system formalism, 4) model reduction or aggregation of multi-fidelity object levels, and 5) the mapping scheme for system states, transitions, and data transformation.

In addition to inherent characteristics of a manufacturing supply chain system from a SoS perspective, another key requirements for domain modeling should include: 1) the simulation objectives and scope, 2) system-related environs and simulation study hypothesis, 3) supply chain constituent sub-systems of interest, and 4) focused system-wide variables. For the manufacturing supply chain system in this paper, the following requirements for each category can be identified and summarized in Table 1.

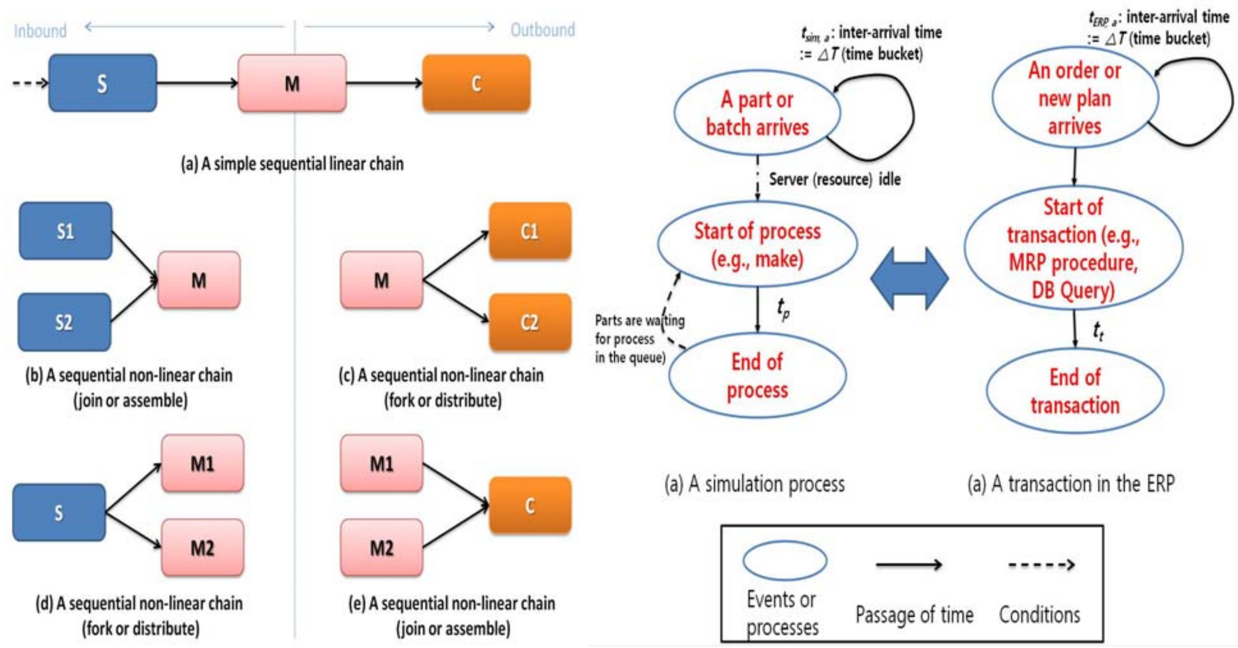
Table 1: Key requirements and scopes identified from the general problem domain analysis

Requirement aspects from the general analysis	**Key requirements and the scope of the problem domain**
The system software modeling objectives and scopes	• In order to produce and deliver finished products to end customers in the most cost effective and timely manner (a typical example of the SC business process scenario) • Evaluation of behavior in physical supply chain execution by SCOR level 1 plans and their policies for the entire federation (level 1: Mid-term) • Evaluation of the each supply chain SCOR level 2 plans for functional processes and their policies in each federate (level 2: Short-term)
The simulation study hypothesis	• MRP/ERP type planning systems are used to coordinate and synchronize various interactions and flows between highly complex variable manufacturing supply chain systems. These same MRP/ERP software models can also serve to coordinate and synchronize complex highly variable simulation models of these same systems as decision models.
Scope of the example Problem domain	• Spatial value chain: Three generic aggregated stages (Suppliers – Manufacturers – Customers) • Horizontal coordinated linkages - External chain: generic buyer – supplier relationship - Internal chain: relationship between functional units (i.e., purchasing/sourcing, production, and sales/delivering) and different control stages (i.e., by order decoupling points) • Hierarchical fidelity/decision making level covered: 2 Levels
Decision making levels with respect to planning processes	• Tactical: Master plan – Resource plan,/Mater Production Schedule (MPS) – Rough Cut Capacity Requirement Plan (RCCP) • Operational: Material Requirements Plan (MRP) – Capacity Requirement Plan (CRP)
The key decision variables	• Inventory levels or order quantities for each federates (for each planning level) ($I_{r_i,\Delta T_n}, x_{r_i,\Delta T_n}, y_{r_i,\Delta T_n}$) • Start and end time (date) (for scheduling level) (St_i) • The size of the efficient time buckets (ΔT_i)
System uncertainties concerned	• Delay time for each process (sojourned time Δ at each state) • Demand variability ($\tilde{d}_{\Delta T_i}$)
Non-anticipatory behaviors of the states to be considered	• State variables observed directly from the aggregate resource or process states that violate the predefined set of cost constraints and policies • Sojourned time of the given processes that violated the time constraints in the given time bucket

3 DISCRETE PROCESS ORIENTED SUPPLY CHAIN MODELING AND SIMULATION

Consider a demand-driven sequential manufacturing supply chain system with three stages (suppliers(S) – manufacturer(M) – customers(C) (Roder and Tibken 2006; Rabelo et al. 2007)). Various instances of this sequential chain based on a generic supplier-buyer relationship can be identified as shown in Figure 2. In most cases, a system can be constructed by combining the general structures represented by simple linear or fork/join relationships between stages through functional linkages. It is noted that functional units within a supply chain organization are similar to each other using the general SCOR configuration (SCC 2011).

In process-oriented simulation models, the system can be represented by block diagrams, or system networks, through which entities flow to mimic real system objects. Meanwhile, in event-driven models, the system can be represented by event graphs, which focus on the abstraction of the event rather than on observable physical entities (Sturrock and Pegden 2011). In this research, a simulation environment is proposed to integrate both approaches: (1) Periodic process simulation using the time bucket-based simulation time advancing scheme for process-oriented (controlled by a transaction manager) and (2) discrete event simulation integration (executed by both state and time events).



(1) Various process instances for the generic supplier-buyer (2) Event graph representations

Figure 2: Various instances of the example manufacturing systems and a simple event graph representation for a process and event in a simulation and ERP model

Processes and activities are invoked by a triggered event, and they are usually executed according to event scheduling procedures. Event procedures update the state of the system, schedule other events, and/or cancel events. In general, events in a supply chain organization can come from the following three sources: (1) events related to task status, such as the end of a task or the beginning of a task; (2) events generated by a task, e.g., an event such as "stock partially available" or "out of stock" is the result of the "check availability" task; and, (3) exogenous events which may arrive from other supply chain partners or from the external environment, e.g., new order arrival, inbound shipment delay, import policy change and so on (Liu, Kumar, and Aalst 2007).

The process-oriented and event-driven approaches are closely related to the two principal time increment methods such as the fixed Δt and variable Δt for simulation execution. This research uses the time buckets that are determined by the transaction manager and forces a federation of simulation models to advance their simulation clocks based on the fixed Δt. The variable Δt refers to the simulation time advances with respect to the "state event" whereas the fixed Δt is associated with the simulation time advances with respect to the "fixed time event".

For variable Δt, all distributed federates update their state at different points by the variable Δt since time increments in a simulation depend on the time of the next event. On the contrary, if a fixed Δt method is used in the system, it is consented that all models update their states at a fixed time increment. We only focus on the latter method and propose the mechanism to handle the exogenous state/time events within a fixed Δt so that all the federates reconcile their system states at the same rate according to the common time bucket.

4 THE FORMALISM OF MANUFACTURING SUPPLY CHAIN SYSTEMS OF SYSTEMS

A simulation model is composed of a set of objects, their behavior and relationships. For example, in a supply chain, the main objects include a set of products, resources and subsequent processes. Those objects mapped into various system states can be evaluated using an analysis of process capability. States of each object are characterized by a set of attributes with either fixed or dynamically changing values. The entire system can be formally represented as modularized components of time and resource/process constrained automata comprised of a location/process and an assignment of values to state variables.

4.1 Structural Modeling Formalism for the SoS Components

Definition 1 A SoS federation of supply chain network SoS_{SCN} is defined as follows;

$SoS_{SCN} = \ < sF, Role, F, Coord >$, if the entire *SCN* is centrally coordinated by a master *SC* planner

$SoS_{SCN} = \ < sF, Role, F >$, if there is no central coordinator, where

- sF is a set of manufacturing supply chain federates such that
  $sF = \left\{ sF^k \mid k = a\ supply\ chain\ stage,\ 1,\ 2,\ 3,\ ...,\ K \right\}$ where
- $sF^k = \left\langle sF^k_{oj[r]} \mid o = 1,...,O;\ j=1,...,J;\ r=\bullet,1,...,R \right\rangle$ where
  - o : A distinct organization (root federate) in a SC federation,
  - j : The system granularity level of the SC federation,
  - *[r]*: the rth sub federate (component) within a parent federate
    r=•: a virtual root federate (e.g., a manufacturer containing physical sub facilities)
    1≤r≤R: a physical federate (e.g., facilities or shops within a federates)

(e.g., $sF^2_{1,J[1]}$ can be a machine (atomic resource) in a manufacturer federate at the SC stage 2 with the lowest atomic level j)

- $role:\ sF \rightarrow\ configuration\ role\ type\ =\ \{Supplier\ D\ type,\ Supplier\ SD\ type,\ Supplier\ SMD\ type,$ $Manufacturer\ SMD\ type, ...\}$
  (e.g., $role\ (sF^k_{oj[r]}) = Assembler\ SMD$)
- $F = \left\langle F^M_{uv},\ F^I_{uv} \right\rangle$ is F^M_{uv} and F^I_{uv} represent material and information flow from federate set ***u*** to a federate set ***v***, respectively.
- $Coord$ is a coordination function that manages interrelations among manufacturing supply chain federates which can be defined depending on whether simulation schemes are centralized or distributed. It is defined as $Coord =< Z^{Coord}, \delta^{Coord} >$. It is identical to the root interaction model IM_0 which is in the root level of the system hierarchy.

Definition 2 A manufacturing supply chain federate for a SoS model sF is defined as follows;

$sF = \ < Z, z_0, z_A, \sum, \delta, A, P, \gamma, BM>$, Where

- The non-empty finite state set Z consists of ξ -dimensional vectors z^T . The components $z^{[l]}$, $l=1,2,3,...\xi$, z are in one-to-one correspondence with the models in the federate at a specific time t and they indicate the number of process instances executing the corresponding stage in the considered non-empty finite set of states and $z_0 \in Z$ and $z_A \subseteq Z$ are an initial state of and a set of accepting states, respectively;

- $\sum$ is a set of events which includes a set of input, output, and internal events such that $\Sigma = \Sigma^{+} \cup \Sigma^{-} \cup \Sigma^{0}$, $\Sigma^{0} = \Sigma^{0}_{PM} \cup \Sigma^{0}_{OM} \cup \Sigma^{0}_{IM}$;
- δ: $Z \times \Sigma \to Z$ is a state transition function;
- A is a set of actions (releasing, reporting, translating, etc.) that a federate can perform;
- P is the finite set of physical preconditions for actions each of which element is in the form of $\rho_{\alpha} \in P$ for $\alpha \in A$ a corresponding $\rho_{\alpha} \in P$ exists, where ρ_{α} is a function that returns either true or false;
- $\gamma : Z \times \Sigma \to A$ is a federate action transition function;
- $BM =< PM, OM, IM, TM, \delta^{BM}, \Gamma^{BM}, \Upsilon^{BM} >$ is a set of behavioral model consisting of a planning, an operational, and an interaction model each of which will be defined later.

Definition 3 A material and information flow which is denoted by F is defined as follows;

$$F = \left\langle F^{M}_{uv}, F^{I}_{uv} \right\rangle, \quad \begin{array}{l} F^{M}_{uv} = \left(f^{M}_{uv} \mid u \in U \text{ and } v \in V \text{ for } u, v \subseteq sF \right) \\ F^{I}_{uv} = \left(f^{I}_{uv} \mid u \in U \text{ and } v \in V \text{ for } u, v \subseteq sF \right) \end{array}$$

The flow can be implemented via a series of messages. The message set assigns a unique identifier to each of these matched external event pairs. Basically, each message corresponds to flow of an entity from the source federate to the destination federate. Therefore, each message should carry the necessary information regarding the attributes of the entity and its timing information (e.g., processes in the source and destination, Qty, Part ID, end virtual time in the time bucket). Hence, we define a message set for the flows in a federation as: $M = \{m_{oo'} \mid \forall f_{oo'} \text{ if } o \neq o' \in F \text{ such that } \sigma^{-}_{oj[r]} \xrightarrow[coord]{f_{oo'}} \sigma^{+}_{o'j[r]}\}$

Definition 4 The supply chain aggregate object entity set oE^{e} is defined as: $oE^{e} = \{oE^{1}, oE^{2}, oE^{3}\}$ where oE^{1} is a set of parts (batch); oE^{2} is a set of orders for a pull type entity; oE^{3} is a set of demand for a push type entity. These entities can behave like a dumb token to carry attributes or transaction and act like an agent to represent intelligent behavior or a state at the system snapshot. It is also defined as $oE^{1} \xrightarrow{valuate} V$ and $oE^{1} \xrightarrow{invoke} F^{M}_{uv}$.

Definition 5 At any instant of time, a state of a federate is given by $Z =< z^{T}, v^{T}, t >$, where $z^{T} \in Z$ is a vector of admissible processes and locations in each model, v^{T} is a vector of the valuation of its real variables, and t is the current time. A state $Z =< z^{T}, v^{T}, t >$ is admissible if invariant $z^{t}[v^{t}]$ holds. A state of a planning model can be also represented by $Z^{PM} =< z^{PM}, v^{PM}, t >$; where $z^{PM} = s\hat{P}^{j}_{c}$ @ $lR^{n}_{oj[r]}$ where $s\hat{P}^{j}_{c}$ is an instantiated process components in the SCOR standard vocabulary.

Definition 6 Stocking resources and functional units are the physical (operation model) and logical (planning model) locations that the process occupies or process behaves. $R^{n}_{oj[r]} =< pR^{n}_{oj[r]}, lR^{n}_{oj[r]} >$ Where $pR^{n}_{oj[r]}$ is a physical part stocking location and $lR^{n}_{oj[r]}$ is a logical part related business functional unit.

4.2 Behavioral Modeling Formalism for the Supply Chain SoS

A behavioral model focuses on the control of entities in a system. For this research, a behavioral model provides an integrated view of the different system abstractions or views such as discrete event based/aggregate process-oriented and discrete control/time based control.

4.2.1 A Planning Model (PM) for the Federation of Transaction Managers

The planning model is responsible for perceiving, organizing, and managing the business logics in business transaction processes. Planning processes balance an aggregated demand across a consistent planning horizon at regular periods, e.g., monthly, weekly, and daily, with respect to the planning hierarchy level. Business logic (a collection of business rules) produces a prescriptive directive for the way business experts want to evaluate facts in order to arrive at a conclusion where the conclusion has both value to the business meaning and value to the business. It must have accurate dynamic data reflecting the current states of the system that are grounded in the operational model (simulation model) in order to generate appropriate planning or control decision policies as outputs for the future (to-be) state of the system. This has become core software used by companies to coordinate various facets of information reported by the execution in every area of the business processes. Eventually, the ERP transaction manager behaves as the process simulation does using the business logic and serves as a controller.

4.2.2 An Operational Model (OM) for the Federation of Hierarchical Distributed Simulation

The operational model represents a realistic structure of physical system resources and material transformation/handling processes related to convert raw material to an end product. The dynamics for state changes depend on information from controls/decision policies for model execution by the planning model and any communication scheme by the interaction model. It corresponds to execution processes in the SCOR framework that are triggered by planned or actual demand and changes the state of products.

4.2.3 An Interaction Model (IM) for Coordination

The interaction model serves as an interface or commissioner to describe the exchanges of (time stamped) dynamic system information and command control flows between the operational and the planning models. In order to facilitate effective information exchanges, aggregation/disaggregation calculation or translation/mapping of data between the two models are needed. In the case of executing federations of distributed simulation and transaction managers, synchronization of the simulation clocks and database transactions in each federate and coordination schemes using rule-based preconditions in an interaction model are also important.

4.2.4 A Temporal Model (TM) for the Time Advancing and State Variables

A temporal model can be created using the properties of a hybrid finite automata enhanced with a clock and state variables which can be queried on states in each model and then reset on transitions. To ensure decidability of the problem, the conditions on clocks and other state variables are usually restricted. Using the temporal model, we can add a set of temporal properties to the finite discrete event models such as PM, OM, and IM. This temporal model can be described as a constraint system where the constraints represent the possible flows, invariants, and transitions for process/resource capability analysis. The definitions for the behavioral, planning, operational, interaction and temporal models and the key system variables used for the planning and operational models are shown in Table 2, Table 3, and Table 4, respectively.

Table 2: Formal definitions for the behavioral, planning and operational models

A Behavioral Model

Definition 7 A behavioral model for a manufacturing supply chain federate, denoted by

- $BM = < PM, OM, IM, TM, \delta^{BM}, \Gamma^{BM}, \Upsilon^{BM} >$
- $\Gamma^{BM} : 2^{Z^{TM}} \rightarrow Z^{OM}$ is a function that maps a set of states within a temporal model to a state within an OM .
- $\Upsilon^{BM} : 2^{Z^{TM}} \rightarrow Z^{PM}$ is a function that maps a set of states within a temporal model to a state within a PM.
- $\delta^{BM} : (Z^{OM} \cup Z^{PM}) \times (\Sigma^{OM} \cup \Sigma^{PM}) \rightarrow Z^{IM}$
  $Z^{IM} \times \Sigma^{TM} \rightarrow Z^{TM}$

$$\delta^{BM}(z,a) = \begin{cases} z', \; if\, z \in Z^{OM}, a \in \Sigma^{OM}, z' \in Z^{IM} \\ \quad if\, z \in Z^{PM}, a \in \Sigma^{PM}, z' \in Z^{IM} \\ \quad if\, z \in Z^{IM}, a \in \Sigma^{OM}, z' \in Z^{OM} \\ \quad if\, z \in Z^{IM}, a \in \Sigma^{PM}, z' \in Z^{PM} \\ \quad if\, z \in Z^{IM}, a \in \Sigma^{TM}, z' \in Z^{TM} \\ undefined, \; Otherwise \end{cases}$$

A Planning Model

Definition 8 A planning model for a manufacturing supply chain federate of the level j, denoted by $(PM(sF_j))$ is defined as:

$PM(sF_j) = < \Sigma^{PM}, Z^{PM}, z_0^{PM}, z_a^{PM}, \delta^{PM}, P^{PM}, \gamma_{release}, releasing, \lambda^{PM} >$

- Σ^{PM} is a set of planning model events that are received and sent from/to an operation model through an interaction model of sF_j such that
  $\Sigma^{PM}_{\boldsymbol{interaction}} = \Sigma^{PM}_{+} \cup \Sigma^{PM}_{-}$
- $Z^{PM} = Z^{PM}_{General} (= Z^{PM}_{0}) \cup Z^{PM}_{Interaction}$
- Σ^{PM} is a set of planning model states each of which is described in a form of resource states;
- $\gamma_{releasing} : Z^{PM}_{A} \times \Sigma^{PM} \rightarrow Releasing$ is a action transition mapping function that outputs a "releasing" action to generate a message with parameters in the form of flow *(F)* after arriving at an accepting state of the PM;
- $\lambda^{PM} : Z^{PM}_{j} \rightarrow 2^{Z^{OM}_{j}}$ is a function that maps a set of states within a operation model to a state within a planning model of the same federate sF_j ;
- $\delta^{PM} : Z^{PM} \times \Sigma^{PM} \rightarrow Z^{PM}$ is a transition mapping function
- $\delta^{PM}(Sales\_forecasting@Sales,\; Notify\, forecast)$
  $\delta^{PM}(Aggregate\_Planning@Headquater,\; Notify\, agg.\, plan)$
  •
  •
  $\delta^{PM}(MRP@procurement,\; releas\, production\, order)$

An Operational Model

Definition 9 An operational model for a manufacturing supply chain federate of the level *j*, denoted by $(OM(sF_j))$ is defined as:

$OM(sF_j) = < \Sigma^{OM}, Z^{OM}, z_0^{OM}, z_a^{OM}, \delta^{OM}, P^{OM}, \gamma_{report}, Reportng, \lambda^{OM}, \tau^{OM} >$

- Σ^{OM} is a set of operation model events that are received and sent from/to a planning model of sF_j through its interaction model such that $\Sigma^{OM}_{interaction} = \Sigma^{OM}_{+} \cup \Sigma^{OM}_{-}$
- Z^{OM} is a set of operation model states each of which is described with a part location within a federate and its step of processing;
- $Z^{OM} = Z^{OM}_{0} \cup Z^{OM}_{Interaction}$
- $Z^{OM}_{interaction} = \{Sourcing\_authorize\ @resource1,\ making\_release\ @resource2,\ delivering\_invoice\ @resource3, ...\}$
- $\gamma_{report} : Z^{OM}_{A} \times \Sigma^{OM} \rightarrow Reporting$ is a action transition mapping function that outputs an action in the "Report" set to generate a message in the form of flow (F) after arriving at an accepting state;
- $\lambda^{OM} : 2^{Z^{OM}_{j}} \rightarrow Z^{OM}_{j-1}$ is a function that maps a set of states within a operation model to a state within a operation model of a parent federate sF_{j-1} ;
- $\Sigma^{OM}_{0} = \{Request\ Payment\ (S),\ Release\ Product\ (M),\ Authorize\ Payment\ (D),\ .....\}$
- $\Sigma^{OM}_{interaction} = \{Soucing\ Complete,\ Making\ Complete,\ Delivering\ Complete,\ Soucing\ Start,\ Making\ Start,\ Delivering\ Start,\ ....\}$
- $\tau^{OM} : Z^{OM} \times \{0,1\} \rightarrow \begin{Bmatrix} if\ 0 = Z^{OM}_{it} \\ if\ 1 = Z^{OM}_{mt} \end{Bmatrix}$ is a function that maps OM states into a set of material transformation states or a set of information transaction states;

Table 3: Definitions for the variables used in a planning model and operational model

An Interaction Model	**A Temporal Model**
Definition 10 An interaction model for a manufacturing supply chain federate is defined as: $IM(sF^k_{oj[r]}) =< Z^{IM}, z_0^{IM}, z_A^{IM}, \sum^{IM}, P^{IM}, \gamma_{translate}, Translating, \delta^{BM} >$ • $\sum^{IM}$ is a set of interaction model events that are received and sent from/to a planning model of sF_j that $\sum^{OM}_{interaction} = \Sigma^+ \cup \Sigma^-$ • Z^{IM} is a set of operation model states each of which is described in a form of a part-related state; $Z^{IM} = \{incoming, outgoing\}$ • $\gamma_{translate}: Z_A^{OM} \times \sum^{OM} \rightarrow Translating$ is a action transition mapping function that outputs a "translating" action to generate message in the form of flow (F) after arriving at an accepting state; • Translating is a set of executable functions α_i, which performs the IM actions specified by corresponding events or messages. • $Translating = \{\alpha_i \mid$ *quering, updating, commisionning, self-correcting, interpreting, messaging*$\}$ • P^{IM} is a finite set of preconditions for an IM actions. It is partitioned so that for $\alpha_i \in Translating$ there is a corresponding $\rho^{IM}_{\alpha_i} \in P^{IM}$, where $\rho^{IM}_{\alpha_i}$ is a function that returns either true of false. • $translating: F \rightarrow (\Sigma^0 \cup \Sigma^-)$ is an IM execution function that interprets a information or material flow to an event generating message; $F \rightarrow \sum$ • $reverse\ translating: (\Sigma^0 \cup \Sigma^+) \rightarrow F$ is an IM execution function that interprets an event to a information or material flow; • $\delta^{IM}: Z^{IM} \times \sum_{\text{interaction}} \rightarrow Z^{OM} \cup Z^{PM}$ • $\delta^{IM}(z^{IM}, a) = \left\{ \begin{array}{l} z^{IM}, \ if\ a \in Z^{IM} \\ z^{PM}, \ if\ a \in Z^{PM} \\ z^{OM}, \ if\ a \in Z^{OM} \\ z^{Coord}, \ Otherwise \end{array} \right\}$, $\delta^{IM}(z^{IM}, f) = \left\{ \begin{array}{l} z', \ if\ z \in Z^{IM} and\ z' \in Z^{Coord} \\ if\ z \in Z^{Coord} and\ z' \in Z^{IM} \end{array} \right\}$	**Definition 11** An temporal model for a manufacturing supply chain federate is defined as: $TM(sF^k_{oj[r]}) =< V, Z^{TM}, \sum^{TM}, \Delta', \Delta T_i, B_v, C^{\Delta T_i}, \psi >$ • $V = \{v_1, v_2, ...v_n\}$ is a finite set of n variables that models the observed continuous or discrete dynamics of the automaton with respect to time and state variables. • $\Delta = \{\Delta_d \mid d = 1,2,...,D\}$ is a valuation set of real number that represents time and D is the wall clock time dimension of TM. This can be used for any real time simulation. • $\Delta' = \{\Delta'_t \mid t = 1,2,...,T\}$ is a set of primed variables which represents values at the conclusion of discrete change. It can be used with a fixed integer unit time increment and this virtual time of each simulation process is independent from the above real-time. It is simply a counter, which is incremented as state events occur during simulation in the variable Δt and as time events occur during simulation in the fixed Δt. • Z^{TM} is a set of states of a temporal model in sF_j; $Z^{TM} = \Delta'\ @Vclock\_ID$ • $\sum$ is a set of events such that $\sum^{TM} = \sum^{TM}_{-} \cup \sum^{TM}_{+} \cup \sum^{TM}_{0}$; • $\psi: Z^{TM} \rightarrow C^{\Delta T_i}$ is a function that maps each state to a set of conditions corresponding to that state; • $\eta^{TL}: 2^{\Sigma^{TL}} \rightarrow C^{\Delta T_i}$ is a function that maps a event set to a set of conditions corresponding to that event; • ΔT is a parameter "length of time bucket" given by a planning model. • $C^{TL}: \begin{array}{l} Z^{TL} \times \Delta' \rightarrow \{true, false\} \\ Z^{TL} \times V \rightarrow \{true, false\} \end{array}$ is a set of constraints expressing conditions on the values of clocks or integer variables; $Z^{TM} = \{Set\ Clock,\ Time\ Elapsed\}$ • $\sum^{TM-} = \{Set\ Clock\_ok,\ Terminate\ Clock\_ok\}$ $\sum^{TM+} = \{Set\ Clock,\ Terminate\ Clock\}$ $\sum^{TM0} = \{proceding\}$ $z_0^{TM} = (ClockID@0.00)$ $z_A^{TM} = \{z_0^{TM}, z_{elapsed}^{TM}\}$

Table 4: Definitions for the variables used in a planning model and operational model

A Planning Model	**An Operational Model**
Definition 12 Planning decision variables (V_d^j) and parameters for the planning models (input variables for the operational model) include: • $\tilde{x}_{R^n_{oj[r]},\Delta T_i}$: Quantity of items to be produced with a functional unit $R^n_{oj[r]}$ in the federate $sF^k_{oj[r]}$ during a time-bucket ΔT_i (available at the beginning of the time-bucket) • $\tilde{y}_{R^n_{oj[r]},\Delta T_i}$: Quantity of items to be transported from a functional unit $R^n_{oj[r]}$ in the federate $sF^k_{oj[r]}$ during a time-bucket ΔT_i (available at the beginning of the time-bucket) • $C_{R^n_{oj[r]},\Delta T_i}$: Capacity of a resource $R^n_{oj[r]}$ in the federate $sF^k_{oj[r]}$ at the beginning of a time-bucket ΔT_i • $\tilde{Ii}_{R^n_{oj[r]},\Delta T_{i+1}}$: The target input inventory level of a resource $R^n_{oj[r]}$ in the federate $sF^k_{oj[r]}$ for the next time bucket from a transaction manager • $\tilde{Ii}_{R^n_{oj[r]},\Delta T_{i+1}}$: The target output inventory level for the next time bucket from a transaction manager • ΔT_i : The predefined time-bucket (duration) for each term i (1 day, 1 week or 1 month) (*i = 1, 2, 3, …, n (n = planning horizon)* **Definition 13** Input variables (V_e^j) for the planning model from the *stage k+1* to the stage k : $\tilde{d}_{R^{n,K+1}_{oj[r]},\Delta T_i}$, $d_{R^{k+1}_{oj[r]},\Delta T_i}$: The forecast and actual demand for the resource $R^{k+1,n}_{oj[r]}$	**Definition 14** System state variables (V_s^j) for operational models • $Ii_{R^n_{oj[r]},\Delta T_i}$: The input inventory level of a stock location $R^n_{oj[r]}$ in the federate $sF^k_{oj[r]}$ at the beginning of a time-bucket ΔT_i at the level *j*, stage *k*, and organization *o* • $Io_{R^n_{oj[r]},\Delta T_i}$: The output inventory level of a stock location $R^n_{oj[r]}$ in the federate $sF^k_{oj[r]}$ at the beginning of a time-bucket ΔT_i • $x_{R^n_{oj[r]},\Delta T_i}$: Quantity of items produced by a set of Z^{OM} resources $R^n_{oj[r]}$ in the federate $sF^k_{oj[r]}$ during a time-bucket ΔT_i (available at the end of the time-bucket) • $y_{R^n_{oj[r]},\Delta T_i}$: Quantity of items transported from a resource $R^n_{oj[r]}$ during a time-bucket ΔT_i (available at the end of the time-bucket)

5 SUMMARY AND FUTURE RESEARCH

In this paper, we presented a fundamental representation of system objects and their relationship based on different model layers of simulation, interfaces, and a transaction manager for value chain decisions. In order to represent system components with descriptive modeling techniques, an architecture consisting of a federation of distributed simulation models and transaction managers that can act as control execution mechanism is introduced in this paper.

Additionally, this paper also provided the functional requirements of the problem domain in order to configure and model virtual and physical system entities for SoS modeling. Based on different viewpoints of the system states, this research illustrates fundamental system resources and objects and how they can be related and mapped into the process oriented supply chain models. We show that system coordination can be achieved by either proactive strategies or mitigation actions through information sharing and exchanges between each functional linkage through federates. The formal representation for the model layers is provided using the modular resource statecharts based on the different viewpoints of system states. Based on the representation, a set of experiments for multiple manufacturing scenarios will be conducted in order to show validity of the proposed methodology and system mapping in the subsequent papers.

REFERENCES

Liu, R., A. Kumar, and W. van der Aalst. 2007. "A Formal Modeling Approach for Supply Chain Event Management." *Decision Support Systems* 43:761-778.

Rabelo, L., H. Eskandari, T. Shaalan, and M. Helal. 2007. "Value chain analysis using hybrid simulation and AHP." *International Journal of Production Economics* 105 (2):536-547.

Roder, A., and B. Tibken. 2006. "A methodology for modeling inter-company supply chains and for evaluating a method of integrated product and process documentation." *European Journal of Operational Research* 169 (3):1010-1029.

SCC. 2011. Supply Chain Operations Reference Model (SCOR) 11.0. Supply Chain Council.

Shin, D., R. A. Wysk, and L. Rothrock. 2006. "A formal control-theoretic model of a human-automation interactive manufacturing system control." *International Journal of Production Research* 44 (20):4273-4295.

Smith, J. S., S. B. Joshi, and R. G. Qiu. 2003. "Message-based Part State Graphs (MPSG): A formal model for shop-floor control implementation." *International Journal of Production Research* 41 (8):1739-1764.

Son, Y.-J., S. B. Joshi, R. A. Wysk, and Jeffrey S Smith. 2002. "Simulation-based shop floor control." *Journal of Manufacturing Systems* 21 (5):380.

Sturrock, D. T., and C. D. Pegden. 2011. Recent innovations in Simio. In *Proceedings of the 2011 Winter Simulation Conference*, Editied by S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu,52-62. Piscataway, NJ, USA: IEEE.

Zeigler, B. P., H. Praehofer, and T. G. Kim. 2000. *Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems*. 2nd ed. San Diego: Academic Press.

AUTHOR BIOGRAPHIES

SEUNGYUB LEE is a doctoral candidate in the Department of Industrial and Manufacturing Engineering at Pennsylvania State University, University Park. He received his B.S. degree in Industrial Systems Engineering from Yonsei University, Seoul, Korea and his M.S. in Industrial and Manufacturing Engineering from Pennsylvania State University. His research interests are simulation-based manufacturing control, and formal modeling of a SoS. His email address is seungyub@psu.edu.

RICHARD A. WYSK is a Dopaco Distinguished Professor in Industrial and Systems Engineering at North Carolina State Univertiy, Raleigh, NC. Dr. Wysk has coauthored six books including *Computer-Aided Manufacturing*, with T.C. Chang and H.P. Wang -- the 1991 IIE Book of the Year and the 1991 SME Eugene Merchant Book of the Year.. He has also published more than a hundred and fifty technical papers in the open-literature in journals including the *Transactions of ASME*, the *Transactions of IEEE* and the *IIE Transactions*. He is interested in: lean manufacturing (waste elimination and setup reduction), product/process engineering, Computer-Aided Manufacturing, Flexible Manufacturing Systems (FMSs) planning, design and control, and most recently the engineering and manufacturing of medical products, including regenerative medical products. His email and web addresses are <rawysk@ncsu.edu> and < http://www.ise.ncsu.edu/wysk/research/index.html>, respectively.

DONGMIN SHIN is an Associate Professor in the Department of Industrial and Management Engineering at Hanyang University, Ansan, Korea. He received the BS and the MS degrees in industrial engineering from Hanyang University, Seoul, Korea, and Pohang University of Science and Technology (POSTECH), respectively. He earned the PhD degree in the Department of Industrial and Manufacturing Engineering at the Pennsylvania State University, University Park. His research interests include information system design, discrete event system control, human-automation interaction systems, and computer integrated manufacturing. His email address is dmshin@hanyang.ac.kr.