ADDRESSING THE OPERATIONAL CHALLENGES IN THE MANUFACTURE OF ADVANCED MATERIALS AND PERFORMANCE PRODUCTS

John M. Wassick*, Anshul Agarwal, Naoko Akiya, Jeff Ferrio and Scott Bury, Fengqi Yu[1]
The Dow Chemical Company, Midland, MI 48674
[1]Northwestern University, 2145 Sheridan Road, Evanston, IL 60208

Abstract

The challenges of supplying advanced materials and performance products are very different from those associated with commodity chemicals. Yet these challenges are well addressed by enterprise-wide optimization techniques. This paper discusses the nature of advanced materials and some challenges and solutions related to product design, production scheduling, process reliability, and logistics. Ongoing research in the use of multi-agent systems for batch process management is also reviewed.

Keywords

Discrete event simulation, Multi-agent systems, Resource Task Network, Crew scheduling

Introduction

The chemical industry is a very mature industry that has undergone significant consolidation and changes in the face of increasing competitive pressures. Long standing manufacturers have been forced to respond to eroding profit margins by making significant changes to their business portfolio and customer base. Chemical companies are de-emphasizing or abandoning historically core businesses and moving downstream to produce components, devices and systems enabled by materials innovation delivered as high margin solutions. The Dow Chemical Company has a strategy built around joint ventures and acquisitions, emerging markets and performance products and advanced materials (Dow, 2010). Companies like Eastman Chemical (Eastman, 2011) and Bayer (Bayer, 2011) are also pursuing higher margins in this manner. For Dow this transformation introduces new operational challenges that are much different from those of commodity chemicals but they still are a target rich environment for process systems engineering.

The aim of this paper to describe the operational challenges in the manufacture of advanced materials and performance products and to show where and how process systems concepts can be used to address those challenges. First to be examined will be key characteristics of developing, manufacturing and supplying these differentiated products. This will be followed by a review of several challenges and their solutions. The first solution is the design of a formulated product through numerical optimization. The goal here is to more quickly and more completely identify candidate products that meet new customer requirements. We will then review the problem of crew scheduling in a job shop where the challenge of scheduling technicians on machines is key to maximizing throughput of the system. This will be followed by discussion of a generic approach to discrete event simulation that accelerates model development. The simulator, having a fixed model structure, relies on capturing the details of the simulated system in the data base supporting the model. Next to be examined is the

* To whom all correspondence should be addressed

logistical problem of optimally loading a semi-trailer with various packages. Finally current research on improving the responsiveness of batch processes with intelligent software agents for batch process management will be described.

Advanced materials and performance products

While advanced materials are not new product lines for Dow, their prominence as a source of profitability certain is new. The market for these products tends to be much closer to the consumer if not serving the consumer directly. These products are also targeted for emerging markets. Here emerging markets means both markets in emerging economies and markets of emerging technologies like solar or wind power. Being first to market for a new technology market is a huge competitive advantage because the customers are generally not willing to change suppliers quickly. Supplying emerging economies with established products can also require agility because a different culture can produce different consumer preferences requiring modifications to products currently sold in established markets. Consequently the objectives of these businesses are to be fast to market with new products, flexible to changing market conditions and customer requirements, and agile with processing configurations supported by complex operating procedures. The section on product design using numerical optimization addresses some of these issues.

Customer expectations

Customer expectations of advanced materials are much more complex than commodity chemicals. The most obvious difference is that these products are highly customized for each customer's application. The key quality metric is the performance of the product in the customer's application not the purity of the product or even the material properties. Therefore standard analytical measurements of the product prior to shipping do not satisfy the customer's quality requirements. Quality control programs demanded by the customer can involve complicated testing regimes that try to mimic the conditions of the customer's application and the customer can also demand detailed process data collection. Advance materials must satisfy the demanding needs of customers with high expectations; however these customers are willing to pay a higher price for differentiated products.

Significant Manual Operations

Manufacturing challenges frequently encountered are low volume raw material additions that are not easily or economically automated, complicated sampling and testing, complicated packaging, and in the case of discrete part manufacturing, sophisticated machine operation. All these challenges lead to significant manual operations. Frequently the manual operations become rate limiting rather than process unit operations as is the case in the production of commodity chemicals. These issues will be explored in the subsequent section on crew scheduling.

Product Mix Complexity

Because advanced materials are customized for each customer's application, the number of products produced in a single facility can reach the hundreds. In addition, new products are introduced very frequently sometimes as often as weekly. Product mix of this complexity requires a very flexible manufacturing operation which most often is accomplished in a batch or a discrete process. If a continuous process is viable it is usually operated in short campaigns that mimic a batch operation. Regardless of the nature of the process, the expansive product mix introduces many challenges such as complicated, time consuming product transitions that if not managed properly severely affect the capacity of the process. This is especially the case for products that are produced in low volume campaigns which require frequent product transitions. The product transitions can be further complicated when there exist product dependent processing routes, i.e. the process is multi-purpose. Product mix complexity also impacts logistics which is examined in the section on optimal loading of semi-trailers.

Unique Plant Designs

Aggressive development and marketing of advanced materials can lead to a very heterogeneous product portfolio with products manufactured in one-of-a-kind processes. If the product serves an emerging technology market then some of the unit operations or machines can also be one-of-a-kind or at least very new with a limited track record of performance. These circumstances lead to uncertainty in plant reliability and yield and therefore plant throughput, which in turn leads to significant technology and business risk. Mitigating the risk while ensuring speed to market can be a key element in a successful launch of an advanced material. The generic discrete event simulator described below is designed to address these issues.

To summarize, speed and flexibility are crucial attributes for successful participation in markets demanding advanced materials. These attributes are less important in traditional commodity chemical markets where low cost to serve and product consistency are much more important. This shift in emphasis has implications on the challenges to be taken on by Process Systems Engineering. Some of these challenges will be discussed in the following sections.

Product Design using Numerical Optimization

Advanced material products in the form of components, devices and systems present enormous challenges for product development, especially for research organizations historically centered on chemistry R&D. The required speed of development needed to take advantage of sudden market opportunities implies that alternatives to time consuming, traditional experimental approaches must be used to shorten the development cycle. In this section we discuss the use of optimization in conjunction with models that predict product performance to provide a very effective means to more quickly and completely explore the design space for formulated products, and thereby arrive at a suitable design faster than manual manipulations of the models.

A formulation is generally developed by mixing/reacting certain ingredients/compounds. The key design decisions are the selection of ingredients that constitute the product, and the quantity of each ingredient. Such decisions are governed by specifications on physical or chemical properties that the product must meet. The traditional method to develop a product involves performing a plethora of experiments; each experiment conducted with certain ingredients chosen in a certain quantity. In certain situations predictive models may be available to help the formulator pre-select formulations for experimental validation. For the most part this approach is based on the formulator's experience and intuition. If an entirely new product specification is desired it may not be intuitive as to what ingredients should be chosen to meet those specifications, or if multiple formulations can be designed that satisfy desired property targets. We propose solving an inverse problem using numerical optimization. In particular, the problem involves feeding the product specifications or desired property targets to a mathematical optimizer which makes the decisions about what are the suitable ingredients for the product and in what quantity they should be mixed. We briefly describe the approach and its application below.

Mathematical Formulation

The basis of our approach is a mathematical model or set of models that can predict the desired properties of the product from a proposed formulation. That is, for any property y^i a mathematical relationship of the form of Eq. (1) exists to link it to the possible components *c*.

$$y^i = f(c_1, c_2, ..., c_n) \tag{1}$$

These models can either be empirical, statistical models, or they can be based on fundamental chemistry and physics. In general these models are nonlinear.

Numerical optimization is used to minimize an objective function. There are several potential objective functions. One could be the sum of weighted least squares deviations between the desired property targets and those predicted by the models in Eq. (1). Another could be based on a cost function for the components. In any case, the optimization has two classes of decision variables, M_c, a binary variable indicating the existence of component *c* in the formulation, and Wt_c the weight fraction or amount of *c* in the formulation.

Additional constraints can be added to the optimization problem to address the preferences of the formulator. For example, limits both upper, Eq. (2), and lower, Eq. (3) may be enforced on the number of components in the formulation.

$$\sum_c M_c \le \max\_components \tag{2}$$

$$\sum_c M_c \ge \min\_components \tag{3}$$

The weight of components existing in the formulation (M_c = 1) can be bounded, while weight of non-existing components can be forced to zero, by Eq. (4) and Eq. (5).

$$Wt_c \le M_c * weight\_\max \qquad \forall c \tag{4}$$

$$Wt_c \ge M_c * weight\_\min \qquad \forall c \tag{5}$$

Certain components can be forced to be in the formulation by Eq. (6).

$$M_c = 1 \qquad \forall c \in C^{reqired} \tag{6}$$

Certain components can be restricted from the formulation by Eq. (7).

$$M_c = 0 \qquad \forall c \in C^{restricted} \tag{7}$$

Other simple constraints can be added to enforce other formulator preferences. Moreover, if the property models involve disjunctions, convex hull and Big-M reformulations can be utilized to effectively incorporate the models of Eq. (1) in the optimization problem. In addition the optimization can be executed repeatedly to generate a list of alternative formulations, $\{z_1, z_2, \ldots, z_n\}$, if integer cuts are added at each iteration. The cuts use a set of parameters $g_{z,c}$ which are assigned a value of 1 if component *c* is in formulation *z*, otherwise the value is 0. After each optimization the set of formulations is updated together with the set $g_{z,c}$ for the new formulation *z*. Then the following cuts in Eq. (8) are enforced during the next optimization cycle

$$\sum_{c} M_c g_{z,c} \leq \sum_{c} g_{z,c} - 1 \qquad \forall z \tag{8}$$

Example Application

The previously described approach has been deployed for designing the products of a formulated system for which 75 different component choices are available. Typically the formulator seeks to generate around 20 different formulations for further consideration. The solution is programmed in GAMS which calls CONOPT and CPLEX. The problem involves 422 continuous variables, 88 discrete variables and 471 equations. A typical run takes 1 CPU minute to provide 20 formulations on a 1.73GHz machine with Intel Core i7 processor. In one application the approach successfully identified a lower cost, non-intuitive alternative to an existing formulation.

Crew Scheduling in an Industrial Job Shop

Some advanced materials are really components that are manufactured in job shops where products follow very complex routings through manually operated machines. In some cases, the key bottleneck is not the machine availability, but the availability of operators with leads to a crew scheduling problem. The incentive for optimization remains the same: maximize value by sequencing jobs such that we fill as many as possible on time.

We will consider a problem that consists of a standard job shop with up to 10 workcenters. Seven workcenters consist of single machines and the remaining three each consists of two machines that can run in parallel. Each of the jobs, or “orders”, is categorized into roughly 170 different categories or “recipes”. A job consists of making a number of discrete parts, with each job having a fixed setup time and variable time. Each recipe dictates a specified routing through the workcenters. Some jobs require routing through up to seven workcenters. All of the jobs have due dates. It is desired that a no-wait policy be enforced between machines to reduce work-in-progress (WIP) from building up in queues between machines.

About 18 operators are available in each of the three shifts run throughout the 5-day work week. A challenging aspect is that each operator is qualified to run only a subset of the machines. The optimization objective is to sequence the jobs on the machines for the next several days such that the maximum number of jobs is completed on time while abiding by the hard constraints of skilled operator availability and the no-wait policy. Schedules for each machine and operator are desired. Jobs can be split across shifts such that a job can be partially completed by one shift and finished by another.

The Resource-Task Network

Building on the success of applying discrete time RTN scheduling technology to batch and continuous plants, an RTN based scheduling approach was developed for this scheduling problem. The discrete time resource task network model (Pantelides. 1994) consists of two fundamental concepts: resources and tasks. Resources can be materials, equipment, utilities, etc. A task is an operation with a fixed duration that can consume and generate resources. In a discrete time formulation, the length of the time horizon is fixed and is discretized into an integer number of time slots with each slot having the same time duration Δt. Tasks must start and end exactly at the grid points, and will be of a duration that is an integer multiple of the discretization time. Likewise, a task can only interact with resources at the grid points. The balance on the resource level at each grid point in the system is given by Eq. (9).

$$R_{r,t} = R_{r,t-1} + \sum_{k}\sum_{\theta=0}^{\tau_k}\left(\mu_{k,r,\theta} N_{k,t-\theta} + \nu_{k,r,\theta}\xi_{k,t-\theta}\right) + \Pi_{r,t} \quad \forall\ r,t \tag{9}$$

Here $R_{r,t}$ is the level of resource r at time t, which depends on the previous resource level at $t-1$ and any interactions that occurred from tasks k. The summation in Eq. (9) represents two types of resource interactions: integer and continuous.

Integer interactions are modeled by the term $\mu_{k,r,\theta}N_{k,r,\theta}$ and are commonly used to track resources that are discrete entities, such as equipment items, e.g., a reactor. $N_{k,t}$ is an integer variable that represents the total number of occurrences of a task k at time t, while $\mu_{k,r,\theta}$ is a fixed parameter indicating the amount of interaction with the resource r. Subscript θ is an integer index that represents a time duration offset from the start of the task k and ranges from 0 to the length of the task τ_k. Continuous resource interactions are modeled by the term $\nu_{k,r,\theta}\xi_{k,t-\theta}$, and are commonly used to track resources that can have a variable level such as a quantity of a material, e.g., the number of pounds in inventory. The resource interaction quantity is the continuous parameter $\nu_{k,r,\theta}$ and the continuous variable $\xi_{k,t-\theta}$ is called the task extent. The final term $\Pi_{r,t}$ in the resource balance is an external influence on the resource. It is the amount of resource that is transferred into or out of the system. This term might be used to model incoming supplies, outgoing deliveries and equipment availability.

Resources in the system are typically constrained within fixed limits such as the number of available reactors or the size of a tank. The extent of a task is also constrained. For example, a batch production task extent might be limited to a minimum and maximum batch size.

Two-Phase Solution Approach

We used a two phase approach to solve the crew scheduling problem. Phase I consists of an upper-level RTN where each task k completes a particular job as shown in Figure 1 below. This phase explicitly tracks the machine resources and the number of operators required to run those machines. Aggregate constraints are written on the number of operators utilized based on both skill sets and attendance to ensure feasibility of the Phase II RTN, which is an operator assignment problem. In Phase II RTN, the operator requirements from Phase I are read as external transfers (Π) into the RTN which then trigger tasks that assign individual operators.

Phase I RTN

The task shown in Figure 1 fills resource Order 1 (Ord1) which is routed through machines in the sequence 1-3-2. The task consumes exactly one Machine 1 (M1) resource at its beginning, and releases it at the time required to process the quantity of units produced to fill Order1. Machines 3 and 2 are subsequently seized and released in sequence to execute the no-wait policy. In addition, operator resources are created by the task to track requirements. In this example, two operators are required on Machine 1. The operator resource is associated with a machine since not all operators are skilled to run each machine.

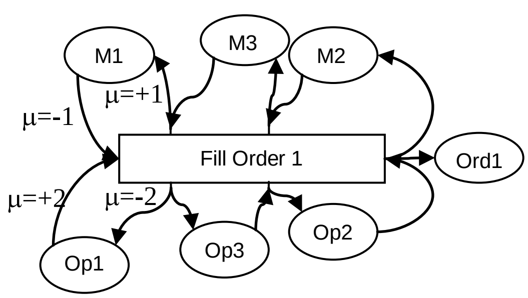

Figure 1. Task to fill orders

Cuts, Eq. (10) are written on the operator resources to ensure that the number of operators required is less than or equal to the total number of operators available. This ensures feasibility in the Phase II assignment RTN (discussed below). $R_{MOi,t}$ is the quantity of machine operators (MO) for machine i at time t. $n_{MOi,t}$ is the total number of operators in attendance at time t that are skilled to operate machine i. $n_{MOiMOj,t}$ is the total number of operators that can run machine i or machine j. Constraints written up to combinations of 3 machines were found sufficient to keep Phase II feasible.

$$
\begin{aligned}
&R_{MOi,t} \le n_{MOi,t} \quad \forall\, t,i \\
&R_{MOi,t} + R_{MOj,t} \le n_{MOi,MOj,t} \quad \forall\, t,i,\, j \ni i \ne j \\
&R_{MOi,t} + R_{MOj,t} + R_{MOk,t} \le n_{MOi,MOj,MOk,t} \\
&\forall\, t,i,\, j,k \ni i \ne j \wedge j \ne k \\
&\vdots
\end{aligned}
\tag{10}
$$

Phase II RTN

The Phase II RTN model assigns individual operators based on the requirements generated in Phase I through resource variable R_{MOi}. In this RTN, the number of operators required at time t is introduced as an external transfer (Π). The maximum level of resource variable R_{MOi} is set to zero which forces one or more assignment tasks to execute. The assignment tasks consume an individual operator and the operator requirement resource, and return the operator at time δ, which is the total time the operator is required which is known from the Phase I output. Figure 2 illustrates an RTN that could assign either T1, T2 or T3 given an external transfer of +1 into the operator requirements resource for machine 1, R_{MO1}.

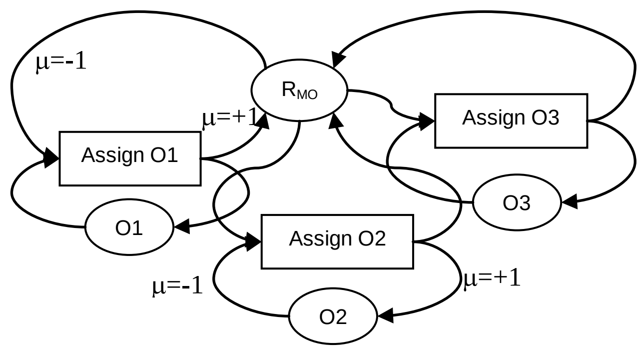

Figure 2. Task to assign operators

Results

The Phase I approach used for scheduling 47 jobs over the course of 38 hours using a 15 minute time discretization results in an RTN with ~6500 discrete variables and 72,000 continuous. A solution with 3% integrality gap is achieved using GAMS/CPLEX in about 300 sec; in about an hour, a solution with 0.07% gap is achieved using an Intel Xeon CPU at 2.13 GHz with 4 cores and 20 GB of RAM. The Phase II RTN consisted of ~139,000 discrete variables and 184,000 continuous. No objective function was specified for the Phase II problem as only a feasible solution was desired; GAMS/CPLEX found a feasible solution in 0.32 seconds. Figure 3 shows an excerpt from the generated schedule showing 4 operators assigned to jobs running on 3 machines. Each individual order is assigned a color to illustrate the linkage between the machines and the operators.

Figure 3. Schedule: Machines & Operators

Rapid Validation of Plant Designs

In the advanced materials sector the pace of bringing new plants online and the uniqueness of their designs presents a significant challenge for design teams to validate that their designs meet business expectations. In general, these expectations involve annual production capacity, overall plant reliability, expected production bottlenecks, capital costs and average inventory levels. Discrete event simulation is a useful tool for analyzing these metrics, especially for plants whose operation can be described as a chronological sequence of events. We use a generic simulation model that applies to any system that can be regarded as a non-steady state process. Examples of process network include batch or discrete manufacturing plant (a network of production units connected by material handling equipment or operator movements) or distribution network (a network of warehouses and distribution centers connected by transportation routes). Although each network is unique, all process networks function by transferring materials from one set of nodes to another set of nodes, with time delays at the nodes and along the arcs. In this paper, we use the term "generic" to refer to a simulation model that can be implemented for a specific system largely through input data alone, with minimal or no changes to the model structure

Our approach is somewhat different from "generic" as flexibility for re-use (Brown, 2010). We do not merely adhered to a philosophy but instead use an actual modeling construct for generic modeling. Our approach also differs from a framework of model generation using a general-purpose modeling language, e.g. SysML (McGinnis and Ustan, 2009: Schönherr and Rose, 2009).

Conceptual Framework

A network is characterized by these elements:
•Individual nodes comprising the network
•Arcs connecting the nodes
•Flow of materials at each arc (quantity and timing)

In our generic simulation the same logic can be used to model any number of nodes with any arbitrary connectivity, whether the flow is continuous or intermittent and event-driven.

Figure 4 illustrates the conceptual difference between non-generic and generic approaches using a hypothetical batch plant with three process steps. The non-generic approach is characterized by logic organized like the network structure. The items simulated are individual batches of products. Queues are used between process steps for work-in-process (WIP) batches. In the generic approach the items simulated are the individual process steps, or nodes in the network. The simulation rules are used to control the timing of release of each item from a single queue, which sets the flow between nodes in terms of the timing of batch execution at each process step. In other words, the network structure is something that emerges when the rules are implemented. WIP is modeled by inventory balance, which allows for straight forward mass balance even when the product batch does not remain intact throughout the network.

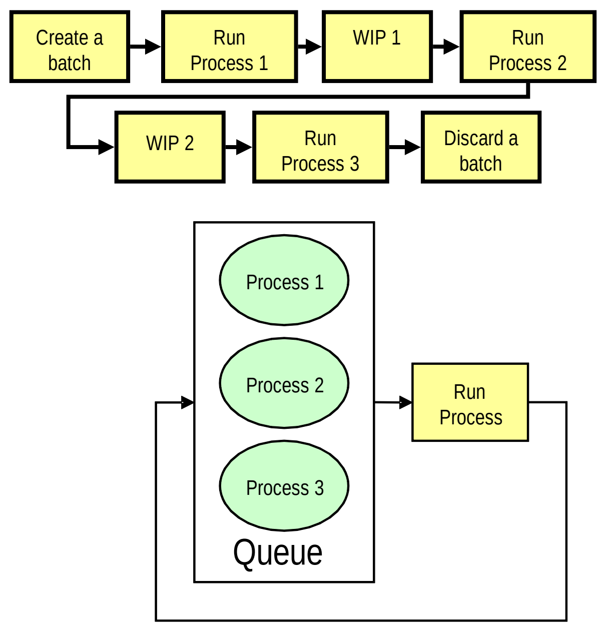

Figure 4. Non-generic (above) and generic (below)

In all simulation approaches found in the literature, the process network is built from objects representing all or parts of a unit operation. When the model is built, however, the connectivity of those objects is fixed. This limitation applies even to so called flexible or general-purpose models, which are essentially model generators (see for example Yuan, Dogan, and Viegelahn, 1993; Selvaraj, Hua, and Hess, 2005). What is unique about our approach is that, because the process network is specified with input data alone, a single model can be used to model any plant by including the appropriate input data file.

Time Dependent Resource Balance

Our approach uses of recipe tables to represent the operation of each node as resource balance over time. At different points in time during a batch, different materials may be consumed or generated, different pieces of equipment may be used, and operators may be required to perform specified steps. These step details are specified in a recipe, which is unique for each product and process.

Our approach uses recipe tables modeled after the RTN described earlier to specify a sequence of steps that represents a unit operation. The timing of steps in a recipe is relative to "fraction of batch complete" which is equivalent to θ parameter. Equipment is seized and released in any step as specified by an equivalent μ. Material used or produced in each step is specified by an equivalent ν related to the batch size (ξ).

Resources included in a RTN can be an abstract entity representing a condition or criterion that needs to be satisfied. For example, a step could generate a "dirty reactor" and a cleaning task could then consume this "dirty reactor" and generate a "clean reactor." A non-zero "inventory" of "dirty reactor" could serve as a rule for triggering this task. In a non-generic formulation, the model structure would need to be extended, using additional blocks and discrete items, to accommodate this cleaning activity.

Data Structure

As the generic simulator is defined by the input data a key component is the database that supports the simulator. This database includes tables of data that specify the plant's production capability: process network, production sequence, batch size, batch time or rate, and resource availability. With these data in the input, it is possible to model very different plants by simply importing a different input database file. There is also a real time data base that contains a set of tables that are used to track various runtime data, which are used in simulation logic execution. Additional data need to be tracked in simulations that include elements such as production planning and scheduling, process failures and other unplanned events, and operator movements inside the plant.

Simulation Logic

Figure 5 shows the high-level simulation logic for running a batch on a process item. Note that additional logic is needed to initialize the simulation and the schedule of batches is assumed given. First, the product ID is assigned to the item. The process items then enter a queue in which they wait until the conditions required for process start are met (resource availability, storage availability, condition of upstream operations). Once the process item is released the process status is updated to, "running."

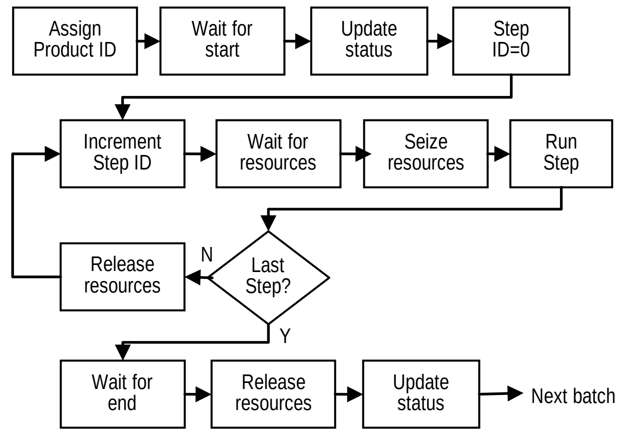

Figure 5. Generic simulation logic

The next part of the simulation logic involves stepping through a batch recipe, in a manner analogous to a real-life batch process control. At each step, the process waits for the availability of resources that are consumed, seize those resources, and run the step. "Wait for resources" is a queue similar to "wait for process start," in that the items are held (rank=null) until all required resources specified in the recipe are available. "Run step" is just a time delay for the duration of the step time. At the end of each step, the resources generated by this step are released. At the last step, however, the releasing of resources takes place only after the process end conditions are met (resource availability, storage availability, condition of downstream operations). Once the batch is completed, the process status is set to "not running."

Application Example

We applied this approach to capital project scoping, during which the capital project team considered different plant configurations.

Figure 6 shows three different configurations that were evaluated, using a single model. In addition, the number of trains per process changed multiple times during the course of this evaluation. The simulation logic was slightly modified to allow any number of trains per process that can run independently of each other, so those changes could also be evaluated with the same model. In spite of the significant changes made to the system under evaluation, the use of a generic simulation model allowed much faster turnaround than would have been possible with a non-generic approach, in which separate models are constructed for different plant configurations.

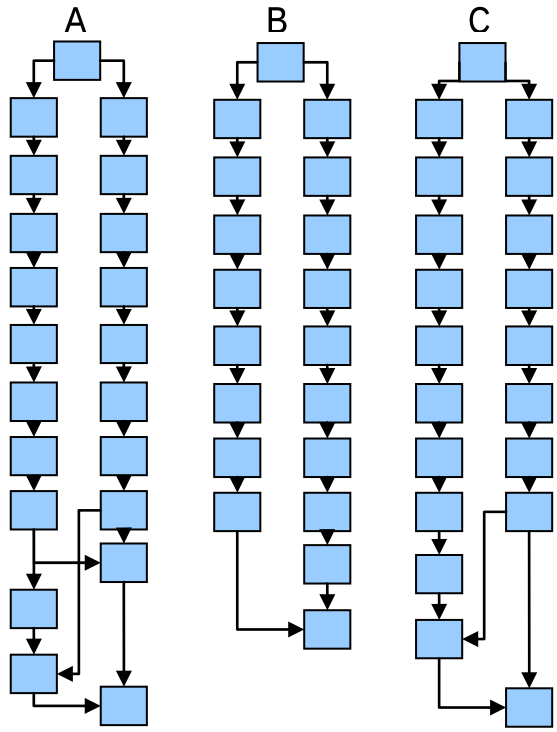

Figure 6. Alternate process designs

Optimal Loading of Semi-Trailers

The problem of loading a semi-truck trailer to its maximum capacity is complicated when the load consists of loose drums and pallets of various products, which is the case for performance products and advanced materials. The loader must distribute the weight so that axle load limits are not violated while also arranging the packages in a secure fashion on the trailer bed to avoid movement as the truck maneuvers over the road. The packages must also be arranged to minimize the time it takes to load the trailer.

A semi trailer is considered fully loaded when either its load consumes the entire floor space of the trailer or the load weight reaches the maximum allowable payload for the trailer. Typically, payload weight is the limiting factor for many of the performance products. The loading situations encountered can vary widely from a load of 80-90 loose drums of a single material, or a similar number of loose drums but spanning several products, or a load of palletized materials, to a load of mixed materials packaged in both drums and pallets. Refer to Figure 7 for an illustration of the acceptable package arrangements.

Drums or pallets of the same material should be placed as close together as possible on the trailer. Rows of drums of the same material should be placed adjacent whenever possible. Drums of different materials can be placed together in three or four drum rows but the number of these mixed rows should be minimized. Likewise, pallet arrangements of the same material should be placed adjacent whenever possible while pallets of different materials can be placed together in the various arrangements. The number of these mixed pallet arrangements should be minimized. For a load picked up at several locations the load pattern must specify that the packages from the first loading dock are loaded first, the packages from the second loading dock are loaded second, and so on. There are several other loading restrictions:

• 3 drum rows must not be placed consecutively
• First and last packages must not be a 3 drum row
• 3 drum rows must be nested between 4 drum rows
• Single pallet must not be the last package
• A single void must consume all unused floor space
• A rear void space is preferred, if placed elsewhere it must be at least 8 feet from the rear

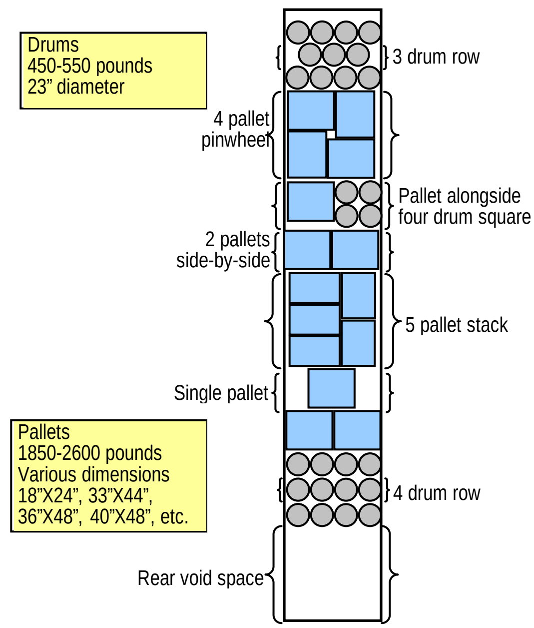

Figure 7. Standard semi-trailer packages

Solving the Loading Problem

We solve the loading problem in a two step optimization procedure. In Step One, all the drums and pallets in the load list are assigned to the predetermined set of allowable package arrangements shown in Figure 7 such that all drums and pallets are accounted for. The goal is to minimize the total number undesired arrangements:

- Three drum row of one material
- Row of drums of multiple materials
- Pallet alongside four drums
- Two pallets side-by-side of one material
- Single pallet
- Four pallet pinwheel of multiple materials
- Two side-by-side pallets of two materials

- Pallet alongside four drum square

This step is executed as a mixed integer linear program minimizing the total number of undesirable package arrangements while satisfying additional constraints, such as: drums of a mixed row must come from the same loading dock and pallets placed alongside four drums must be picked up at the same loading dock as the drums.

The solution is passed the second optimization to determine the position of the package arrangements. The solution here is constrained by the weight distribution across the three axles of the tractor and trailer, the distance between package arrangements of the same material, the location of the void in the trailer, the physical dimensions of the trailer, the location of three drum rows relative to each other, and the pick-up order of the packages that make up each package arrangement.

We solve this step using another mixed integer linear program. In this case, the objective is to maximize the weight loaded on the trailer discounted by a penalty associated with the distance between package arrangements of the same material and a penalty for placing a void space other than in the rear of the trailer. This optimization is subject additional restriction like:

- There must be no more than one void
- A three drum row must be preceded by a four drum row
- The entire trailer floor space must be accounted for
- A package must be loaded before materials picked up at later loading docks and after material picked up at earlier loading docks.
- No void after a three drum row or a single pallet.

Example Load

Table 1 lists a complicated load containing a mixture of drums and pallets containing nine different products picked up at three different loading docks. The load requires the optimizer to perform several difficult loading tasks, such as create and place a mixed drum row and arrange the load to match the pick-up sequence. Figure 8 illustrates the optimized load plan.

Table 1. Example load list

Prod	Package	Qty	Total lbs	Dock
1	Drum	4	2,076	2
2	Drum	5	3,066	2
3	Drum	6	3,672	2
4	Drum	4	1,996	2
5	Drum	12	6,002	2
6	Drum	4	2,025	2
7	Drum	20	10,920	3
8	Pallet 50X43	2	5,160	1
9	Pallet 48X45	4	10,320	1

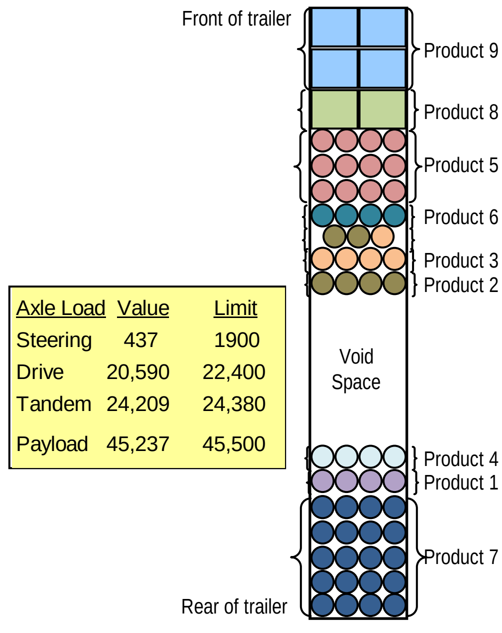

Figure 8. This is the figure caption

Real-Time Batch Management with Multi-Agent Systems

As we stated earlier, key objectives of advanced materials businesses are to be fast and flexible to changing market conditions and customer requirements. These business characteristics require exceptional performance and reliability of batch process management and control. The system must be able to introduce new products, change the product mix, and reroute existing production campaigns. This is possible if we have flexibility to allow rapid changes to recipes, control procedures and production schedules. In addition batch processes must be able to handle unpredictable events from external markets (late order arrivals, orders cancellations, delayed raw material shipments, modifications in order due dates and/or customer priorities) and from the operational level (changes in batch processing/setup times, unit breakdown/startup, reprocessing of batches, changes in resource availabilities).

Effective scheduling of batch processes and associated human resources on a real-time basis is essential to flexible manufacturing. Mixed-integer programming (MIP) approaches have been widely studied for generating optimal production schedules offline (Mendez et al., 2006). However these approaches require significant human intervention when unforeseen circumstances arise or when it is difficult to capture in the model all aspects of the scheduling problem such as the interactions between operators and their decision-making.

Multi-agent systems (MASs) offer an alternative approach by using a coordinated community of autonomous and self-adaptive agents in a distributed and co-operative manner (Jennings & Wooldridge, 1998; Siirola et al., 2003; Lee & Kim, 2008). When changes take place the intelligent agents can be self-adaptive and quickly re-optimize and redirect batch operations. In addition, the interactions between operators and their decision-making, communication and learning processes can be well captured by an agent-based model. The distributed decision-making of the intelligent agents can provide effective decision making with computational times suitable for real-time execution.

We are developing a generic batch management system for real-time operational control of batch processes based on agent-based techniques and distributed optimization strategies. In the following sections we will provide a detailed description of the MAS to be developed and then introduce our proof-of-concept demonstrator focused on the production scheduling tasks.

Agent-Based Real-Time Batch Management System

Our objective is to develop a MAS for real-time operation of batch processes and the associated human resources. In this system, human operators and all tasks related to process control, scheduling and optimization in this system are carried out by autonomous agents that are capable of interacting and negotiating with each other to bid for tasks and make decisions. The system will be designed to be generic, allowing it to be tailored to suit a wide variety of different batch and discrete manufacturing problems. The architecture of the MAS will be defined following ANSI/ISA-S88 and ANSI/ISA-S95 standards. A conceptual structure of this system is depicted in Figure 9.

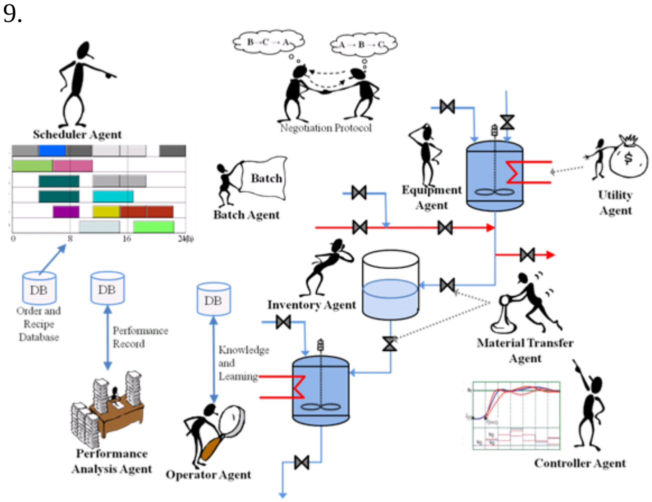

Figure 9. Agent-based real-time batch management system

Due to the self-adaptive and distributed agents, the system is expected to have the following capabilities:

(1) Timely generate optimal production schedule: the proposed MAS may not only increase the visibility of production information and the communication between operators, but also quickly improve the production schedule based on the timely and active production information (e.g., actual start/end time of a batch), batch conditions and operator coordination.

(2) Dynamically optimize and control the manufacturing operations: the MAS can effectively track and guide physical operations according to the operations schedule, and actively control the progress of production operations to meet the planned schedule by performing dynamic optimization to determine the best time-dependent operational conditions.

(3) Real-time performance analysis: the system may evaluate both the effectiveness (e.g., cycle time, on time delivery) of the generated schedules and the performance (e.g., idle rate, inventory level and resource utilization rate) of processing units and operators.

To achieve these objectives, the MAS is expected to consist of task agents, equipment agents, operator agents, utility agents, inventory agents, a material transfer agent, a performance analysis agent, a scheduler agent, a controller agent and the corresponding databases (see Figure 10).

The task agents, equipment agents, operator agents, utility agents and inventory agents are basic batch process control agents defined following ANSI/ISA-S88 standard. Their numbers are the same as the corresponding physical units in the batch plant. For example, if the batch process has 5 pieces of equipments, then the MAS will include 5 equipment agents for each piece of equipment.

The scheduler agent and the controller agent are responsible for advanced decision-making based on solving centralized optimization problems for process scheduling and dynamic optimization, respectively. Although they are not required in a basic MAS for batch scheduling, their presence in the system will potentially improve the optimality of the scheduling and control decisions made by the distributed intelligent agents.

The material transfer agent controls all the valves, pumps and other transfer facilities to move the materials according to the production schedule and tasks.

The performance analysis agent monitors the production activities based on real-time process information. Whenever production status changes (e.g. a batch needs to be re-processed), this agent will alert other agents to adjust the production schedule and/or control trajectories in response to the abnormal event.

There are two databases in the system. One is linked to the performance analysis agent to record all the event of the system and the performance of equipments and operators; the other one is linked to the operator agents for the knowledge and learning of each operator.

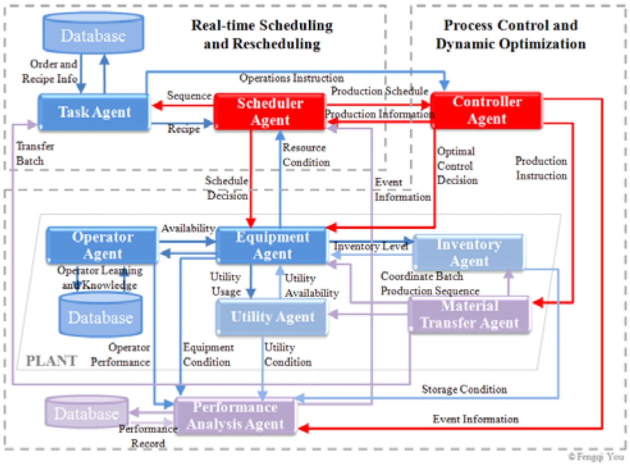

Figure 10. Expected architecture of the MAS

All agents in the MAS are connected to the network independently. Each agent can reach any others in the system by message passing following negotiation protocol. Agents are also aware of the rules in which information is exchanged with other agents. Such rules used by agents for communication and decision-making are known as negotiation protocols. A widely used negotiation protocol is the contract net protocol, which involves four major steps: Task announcement, bidding, awarding, and expediting. A variation of this protocol can be developed to improve the communication efficiency by introducing additional "virtual" currency, i.e. market-based contract net protocol (Kumar et al., 2008).

In one approach that we will investigate, at a chosen frequency, the initial schedule will come from solving a deterministic MIP problem, and subsequent rescheduling will be handled by the aforementioned agent-based approach. Once the schedule is determined, the controller agent, who is responsible for generating the optimal time trajectories of important process conditions (e.g. feed rate, temperature, pressure), will then solve a dynamic optimization problem under fixed schedule to determine the optimal control profiles. Since the schedule is fixed, this dynamic optimization problem can be solved very quickly. After the optimal trajectories are obtained, controller agent will send the updated information to the scheduler agent for confirmation. Upon receiving the "awarding" information from the scheduler agent, the production task will be processed. If the optimal solution of the dynamic optimization problem results in some changes of processing time, changeover time, etc. or a disruption occurs during the period, the pre-optimized schedule becomes invalid and a new schedule is needed. The scheduling agent will have the option to restart the negotiation process or employ some rescheduling strategies based on meta-heuristics and/or mathematical programming to improve the solution quality.

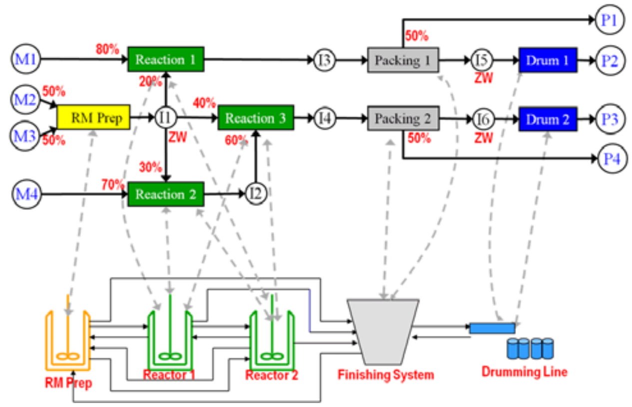

Figure 11. Demonstration problem

Proof-of-concept Demonstrator

A proof-of-concept demonstrator has been developed using the Repast platform (North et al. 2007). The demonstration problem considered is a batch plant consisting of one raw material preparation tank, two reactors, one finishing system and one drumming line. There are eight production tasks. Four raw materials, six intermediate products and four final products are involved in the production. A state-task-network (STN) of this problem (Kondili, et al., 1993) and the mass balance relationship of the tasks are given in Figure 11.

For simplicity, the proof-of-concept demonstrator only includes those basic batch process control agents (task agents, equipment agents, operator agents, utility agents and inventory agents), although the system can be extended to incorporate other agents introduced in the previous section.

Figure 12 is a screen shot when the simulation is running. Operators are on the right, raw material storage tanks are on the top, final product tanks are at the bottom and the intermediate storage tanks are on the left. We can see that each equipment agent has inlet arrows from some inventory agents and outlet arrows to other inventory agents. These arrows reveal the input and output relationships of a batch being processed by the equipment. Based on this relationships and the STN given in Figure 11, one can determine which task is being processed in the equipments, although the task agents are not visible in the GUI. The blue color signals emitted by the equipments represent the production progress. The stronger the signal is the shorter time remains to complete the current batch. We can also see that the operators are moving around the plant to look for tasks. When an operator is working on a certain task, the color changes from white to red, and the color changes back to white when the operator completes the task. The system also displays in real time the inventory levels of all materials, batch sizes in all equipments, and the remaining demand of all final products (see the right of Figure 12 for the screen shots of three dynamic charts).

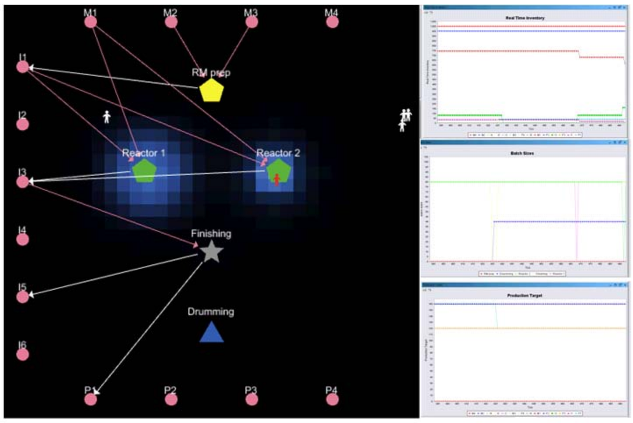

Figure 12. Screen shoots of the demonstrator

The system also allows human-computer interaction in real time. With the buttons on a control the user can add or remove an operator in real time during the simulation. One can also issue a random equipment breakdown and recover the broken-down equipment to check how it affects production activities. Last but not least, random demand fluctuation can be introduced through the GUI.

Comparison with Mixed-Integer Programming Approach

The objective of the demonstrator is to test the concept of applying MAS to a batch operations problem and to build in enough flexibility to allow the method to be applied to a range of similar problems. The use of an agent-based system, which relies on message passing as the mechanism for negotiation, inevitably adds significant overhead to the system in terms of complexity. The asynchronous nature of the competitive bidding process also makes the generation of perfectly optimal solutions unlikely. The main advantage is the ability to react to changes and to adapt the production strategy accordingly. But it is important to have an indication of how good the scheduling capability is so that the impact of improved algorithms can be quantified. To this end we solved the demonstration problem with the mixed-integer linear programming formulation proposed by Maravelias and Grossmann (2003) with the objective of cost minimization and a demand due date of 24 hours. The demand information and the fixed and variable costs of each task are given in Tables 2 and 3. An optimal solution with 0.1% gap requires 4,385 CPU seconds on a IBM T400 laptop with Intel 2.53 GHz CPU and 2 GB RAM. The solution yields a minimum production cost of $139,472. For comparison, the proof-of-concept demonstrator on the same computer with the same input data and the same time horizon took 63 CPU seconds to simulate the operation of filling the orders. The schedule leads to a production cost of $140,672. Thus, the MAS approach generates a feasible production schedule much faster than the MIP approach (63 vs. 4,385 CPUs), although the resulting production cost is slightly higher ($141672 vs. $139,472).

Table 2. Final product demands and due date

Final products	Demand (Due in 24 hours)
P1	160kg
P2	160kg
P2	40kg
P4	40kg

Table 3. Fixed and variable costs of each task

Task	Fixed cost	Variable cost
Rm Prep	$600	$30/kg
Reaction 1	$3,200	$270/kg
Reaction 2	$1,600	$170/kg
Reaction 3	$1,700	$80/kg
Packing 1	$500	$15/kg
Packing 2	$500	$15/kg
Drum 1	$200	$30/kg
Drum 2	$200	$30/kg

The production schedules generated by the two methods are given in Figures 13 and 14. The labels on the batches show the corresponding task name and production amount. For instance, Label "R2:48" implies that the task is "Reaction 2" and this batch produces 48kg of intermediate I2, which is the only product of "Reaction 2". We can see that both schedules yield the same amount of final products, i.e. meeting the production target (see Table 4). However, the schedule generated by the MIP approach has a time span of 24 hours, which is longer than the one for the MAS approach, although the schedule by the MIP approach might have less costs. We should also note that we do not consider travel time of operators (i.e. assuming they are moving instantaneously) in the MIP model, but the MAS approach explicitly accounts for traveling time of operators through modeling the operator behaviors. So the gap time between two nearby processing stages are larger in the MAS results than the ones in the MIP results.

There are pros and cons of both approaches. The advantage of the MAS approach is that it can generate good feasible schedule quickly and the system is adaptive to the changing environment. Although optimality is very useful for emphasizing improved profitability it is hard to fully implement because of the dynamic nature of industrial environments. Therefore, MAS approach can be a complement to the existing research on batch scheduling, which is dominant by MIP approach. The opportunity exists to integrate both approaches to take advantage of their strengths.

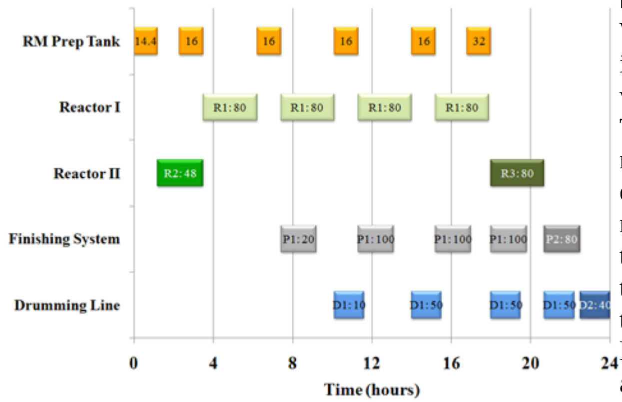

Figure 13. Optimal production schedule obtained with mixed-integer programming

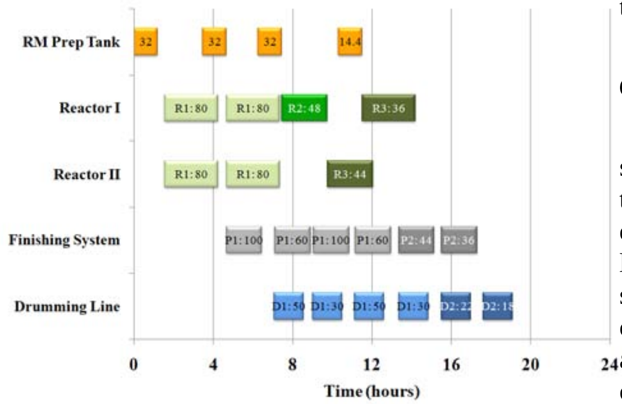

Figure 14. Production schedule obtained with the proof-of concept demonstrator

Table 4. Total production amount of each task (both solutions lead to the same results)

Task	Product	Total production amount
Rm Prep	I1	110.4kg
Reaction 1	I3	320kg
Reaction 2	I2	48kg
Reaction 3	I4	80kg
Packing 1	P1/I5	320kg
Packing 2	P4/I6	80kg
Drum 1	P2	160kg
Drum 2	P3	40kg

Future Directions

Future research will focus on the integration of agent-based techniques and distributed optimization strategies. We will extend the proof-of-concept demonstrator to incorporate a scheduler agent and a controller agent, which both have centralized optimization capabilities. The development of efficient algorithm for inter-agent negotiation scheme to improve the communication efficiency and the development of novel decomposition methods for MIP-based batch scheduling model to ensure the solution optimality are necessary components toward the integration of these two techniques. Another challenge to be addressed is how to model the decision-making and learning processes of operators using intelligent agents and how to integrate these agents into the MAS for batch process operations. We will also investigate how intelligent agents can be incorporated into commercial process control systems. Ultimately real-world applications at The Dow Chemical Company will be used to verify the proposed models and methods.

Conclusion

This paper provided a brief survey of challenges of supplying advanced material and performance products to the market. Certain problems in the areas of product design, production scheduling, process reliability, and logistics were examined more closely and implemented solutions were described. Early stage research in the use of multi-agent systems for batch process management was also presented. This survey suggests that many opportunities exist for enterprise-wide optimization in a manufacturing sector dominated by batch and discrete processes. Thus as major chemical companies turn their attention to new process and product technology, process systems engineering will continue to provide the means for improved profitability.

References

Bayer (2011) "Investor Handout, Boston Roadshow" Bayer AG, Leverkusen, Germany

Brown, N. A.. (2010). "Model Flexibility: Development Of A Generic Data-Driven Simulation." *Pro-ceedings of the 2010 Winter Simulation Conference.*: Institute of Electrical and Electronics Engineers, Inc.

Dow (2010) "Q4 2010 Earnings Conference Call" The Dow Chemical Company, Midland, MI.

Eastman (2011) "Investor Day 2011" Eastman Chemical Compnay, Kingsport, TN.

Jennings, N.R., Wooldridge, M. (1998). Applications of intelligent agents. In *Agent Technology* (Jennings & Wooldridge, eds.). pp 3–28, Springer, Berlin.

Kondili, E., Pantelides, C. C., & Sargent, W. H. (1993). A general algorithm for short-term scheduling of batch operations-I. MILP formulation. *Comput. Chem. Eng., 2*, 211 - 227.

Kumar, N., Tiwar, M.K., Chan, F.T.S. (2008). Development of a hybrid negotiation scheme for multi-agent manufacturing systems. *Int. J. Prod. Econ.*, *46*, 539–569.

Lee, J.-H., Kim, C.-O. (2008). Multi-agent systems applications in manufacturing systems and supply chain management: a review paper. *Int. J. Prod. Res.*, *46*, 233–265.

Maravelias, C.T., Grossmann, I.E. (2003). New General Continuous-Time State-Task Network Formulation for Short-Term Scheduling of Multipurpose Batch Plants. *Ind. Eng. Chem. Res., 42,* 3056-3074

McGinnis, L. and Ustun,V. (2009). "A Simple Example Of SysML-Driven Simulation". *Proceedings of the 2009 Winter Simulation Conference.*: Institute of Electrical and Electronics Engineers, Inc.

Mendez, C. A., Cerda, J., Grossmann, I. E., Harjunkoski, I., Fahl, M. (2006). State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Comput. Chem. Eng., 30*, 913–946.

North, M.J., Howe, T.R., Collier, N.T., Vos, J.R. (2007). A Declarative Model Assembly Infrastructure for Verification and Validation, *In Advancing Social Simulation: The First World Congress* (Takahashi, Sallach & Rouchier, eds.,), Springer, Heidelberg, FRG.

Pantelides, C. C., (1994) "Unified frameworks for optimal process planning and scheduling", *Foundations of Computer-aided Process Operations*, New York: Cache publication, 253-274

Schönherr, O and Rose, O. (2009). "First Steps Towards a General SysML Model for Discrete Processes in Production Systems", *Proceedings of the 2009 Winter Simulation Conference,* Institute of Electrical and Electronics Engineers, Inc.

Selvaraj, R., X. Hua, and T. Hess. (2005) "Model-Centric Method and Apparatus for Dynamic Simula-tion, Estimation and Optimization." U.S. Patent Application Publication US 2005/0071137 A1.

Siirola, J. D., Hauan, S., Westerberg, A. W. (2003). Toward agent-based process systems engineering: proposed framework and application to non-convex optimization *Comput. Chem. Eng., 27*, 1801-1811.

Yuan, Y., C. A. Dogan, and G. L. Viegelahn. (1993) "A Flexible Simulation Model Generator." *Computers and Industrial Engineering* 24:165-175.