

SIMULTANEOUS BATCHING AND SCHEDULING FOR CHEMICAL PROCESSING WITH EARLINESS AND TARDINESS PENALTIES*

MAGED DESSOUKY, BRIAN KIJOWSKI, AND SUSHIL VERMA

*Department of Industrial and Systems Engineering, University of Southern California,
Los Angeles, CA 90089, USA*

176 Charles Street, Wilkes-Barre, PA 18702, USA

i2 Technologies, 303 Twin Dolphin Road, Suite 210, Redwood City, CA 94065, USA

We consider the problem of determining the allocation of demand from different customer orders to production batches and the schedule of resulting batches to minimize the total weighted earliness and tardiness penalties in context of batch chemical processing. The problem is formulated as a mixed-integer nonlinear programming model. An iterative heuristic procedure that makes use of the network nature of the problem formulation is presented to approximate an optimal solution. An algorithm polynomial in the number of batches to produce is also presented that optimally solves the problem under special cost structures.

(CHEMICAL PROCESSING; BATCHING; SCHEDULING; WEIGHTED EARLINESS/TARDI-
NESS)

1. Introduction

Chemical production lines in the past were typically dedicated to one or a few products. These production lines followed a continuous process, producing one product at a time. A continuous flow production line was the most efficient manner to operate a chemical plant when it was required to produce only a few high-volume products. However, the market pressure for greater product variety necessitated a shift to more flexible production lines that could manufacture a greater variety of chemical products. As a result, there has been a gradual shift from continuous processes to batch processes in the chemical industry (Ku, Rajagopalan, and Karimi 1987; Rippin 1991). Batch chemical processing is characterized by the production of discrete quantities of chemicals on multipurpose equipment. The batch process approach to chemical processing is effective at economically elevating product mix and responding to varying product quantities.

This trend in the chemical process industry to operate in batches parallels the shift to small lot-sizes in discrete-parts manufacturing. However, there are several distinguishing features between batch chemical and discrete-parts manufacturing. Most prominently, for many chemical processes (e.g., mixing, reaction, distillation), the operation processing time is

* Received September 1996; revisions received March 1997, December 1997, and July 1998; accepted October 1998.

independent of the size of the production batch, unlike most discrete-parts manufacturing processes. For example, the operation processing time at a mixing tank is a function of the ingredients and the equipment resource and does not depend on the size of the batch. Hence, in chemical processing the batch size is not constrained by the operation processing time. Instead, it is constrained by the equipment capacity and the desire to keep inventory at a minimum. In addition to the standard savings in interest expense of maintaining low inventories, small inventories limit (1) the amount of spoilage, since some chemical properties such as molecular composition have stability time constraints after an operation, and (2) the high investment cost in purchasing storage tanks.

In addition to keeping inventories at minimum, surveys (Chaudhary 1988) show that customer service (i.e., meeting the due dates of customers) is an important concern in the chemical process industry. Ku and Karimi (1990) suggest using the total weighted tardiness criteria as a surrogate measure for customer satisfaction. For this reason, we will consider both tardiness and earliness costs in this paper. The main distinction between problems with tardiness penalties only and those with both earliness and tardiness penalties is that in the latter problem it may be necessary to insert idle time before the start of processing a batch to avoid earliness costs, whereas in the former inserted idle time will only expose the batch to tardiness penalties without any savings in earliness penalties.

The focus of this paper is to study the tradeoff between the earliness penalty and tardiness penalty on the joint decision problem of *batching* and *scheduling* for a single-stage chemical processing environment with fixed batch sizes, sequence-independent setup times, and identical processing times. With *fixed* batch sizes, all batches are of the same size for a given period of time, as opposed to a *nonfixed* batch size environment, in which each batch has its own size. The setup times are assumed to be independent of the sequence. The term *stage* refers to a processing step in the plant. The batching (allocation) decision determines which customer orders a production batch satisfies. The batching decision must not only consider the due dates of the customer orders but the cost of being either early or tardy for each customer order. With both earliness and tardiness penalties, it may no longer be optimal to group demand in the rank order of their due dates. Since a batch can satisfy the demand of multiple orders, a batch can be said to have *multiple* due dates. However, which due dates are served by a particular batch forms part of the decision variables. Only after solving the problem can we associate due dates with a batch. In this sense, it would be misleading to call this problem a multiple due-date problem. The batching decision is especially important when there are many customer order sizes smaller than the batch size, since a particular batch will satisfy the demand of many orders. The scheduling decision determines when the batch is produced. Given an allocation, the batch schedule must take into consideration the different due dates of the various customer orders that the batch satisfies and their associated earliness and tardiness costs. In a review paper on this integration problem as related to discrete-parts manufacturing, Potts and Van Wassenhove (1992) argue that these decisions need to be made jointly and are interrelated. Moreover, the authors argue that with the advent of computer integrated manufacturing, these decisions will be integrated and computer-controlled.

We point out that there has been a significant amount of work on single-machine scheduling with both earliness and tardiness penalties (see Baker and Scudder [1990] for an excellent review of this work). However, there is no similar body of work that considers both earliness and tardiness penalties in context of the joint batching and scheduling problem in batch chemical manufacturing. Individually, the batching and scheduling problems in our batch chemical manufacturing context can be solved easily using existing methods. However, the joint problem is not amenable to the standard methods. Sections 2 and 3 provide the background for the problem. In Sections 4 and 5, we formulate the problem as a mixed-integer nonlinear programming model and then present an efficient iterative heuristic procedure that makes use of the network nature of the formulation to approximate an optimal solution. We show that the problem may be solved in polynomial time under specialization

of the earliness and tardiness penalties and how the specialization can be used to calculate the lower bound for a general instance of the problem. Finally, we compare the proposed iterative heuristic against two single-pass heuristics and benchmark it against both the lower bound presented in Section 6 and the optimal solutions generated by CPLEX, a commercially available optimization software package.

2. Practical Motivation

Our particular motivation in studying the interrelationship between the batching (allocation) and scheduling decision rules in batch chemical processing arose from a study of a proposed pesticide plant in the Middle East. The proposed pesticide plant can be classified as a single-stage batch chemical processing environment with fixed batch sizes, sequence-independent setup times, and identical processing times across all batches. The proposed pesticide plant was expected to produce 40 different types of pesticides in six different mixing tanks. Each product had a fixed assignment to a mixing tank. All product types had the same mixing time of 8 hours (one shift). The mixing time included the setup time and was independent of the batch size. Because of the high cost of reconfiguring the tank (e.g., changing the valves), the batch size was considered fixed throughout the planning horizon (in the order of several months); that is, all products at the same mixing tank were produced in equal batch sizes. With fixed batch sizes, all batches are of the same size for a given period of time as opposed to a non-fixed batch size environment in which each batch has its own size. Besides our pesticide application where the high cost of reconfiguring the tank necessitated a fixed batch size, Selen and Heuts (1990) give examples of batch chemical processing environments in which the batch sizes are held fixed. One application that they present is latex production, where the batch sizes at the reaction operation are fixed because of the preassignment of capital-intensive tanks with limited storage capacity.

Since the proposed plant was in the planning stage, there was no actual data on operating costs. Estimates for inventory holding costs per unit time depended on the product type and varied from a low of \$.01 per liter-day to a high of \$.27 per liter-day. Planners for the pesticide plant could not provide estimates for tardiness costs but felt they would at least be as high as the earliness costs. Forecasted demand per order varied from a low of 1,000 liters to a high of 20,000. Batch sizes of 5,000 liters and 10,000 liters were considered. The selection of the batch size dictates the level of batching and capacity within a plant. With a batch size of 10,000 liters, there would be a significant number of orders smaller than the batch size, resulting in more batching. The plant capacity is increased, with the larger batch size resulting in more slack in meeting the due dates. The advantage of the smaller batch size is fewer levels of inventory. In the experimental section, we test the model under scenarios with varying levels of batching and slack in the due dates to reflect the different batch sizes that need to be considered in the proposed plant.

3. Related Work

Reklaitis (1991) notes that there has been a significant amount of research on batch scheduling, owing to its practical importance and theoretical challenging nature. The work has either focused on developing mathematical programming formulations or developing heuristic algorithms for various special cases of the problem (Pekny, Miller, and McRae 1990).

Mathematical programming formulations of the joint batch-sizing, batching, and scheduling problem with nonfixed batch sizes include the optimization models developed by Faqir and Karimi (1989), Birewar and Grossman (1990), Kondili, Pantelides, and Sargent (1993), and Dessouky, Roberts, Dessouky, and Wilson (1996). These models do not consider the constraint that all batch sizes must be equal since they are for a nonfixed batch size environment.

Although prior optimization models by Smith-Daniels and Ritzman (1988), Patsidou and Kantor (1991), and Heuts, Seidel, and Selen (1992) have addressed sequencing in a more general fixed batch size problems (nonidentical batch processing times and sequence-dependent setup times), they do not take into consideration the batching decision. That is, they implicitly assume that orders of the same product type have the same earliness and tardiness penalties. The batching decision is of special importance when there are many customer order sizes smaller than the batch size with varying earliness and tardiness penalties.

Dessouky and Kijowski (1997) consider three decisions (batch-sizing, batching, and scheduling) for sequence-independent setup times and identical processing times. Their model only considers earliness (inventory) costs and assumes that all orders will be met by their due date and all demand ordering the same product type have the same earliness penalties. This research extends this work by including tardiness penalties. Note with both earliness and tardiness penalties, it is no longer optimal to group the demand based on the rank order of the customer due dates.

4. Problem Statement

We formulate the joint batching and scheduling problem for a batch chemical processing environment with sequence-independent setup times and identical batch processing times as a mixed-integer nonlinear programming model. The problem is to determine (1) the optimal allocation of demand from customer orders to batches and (2) the schedule of batches to minimize the total weighted earliness and tardiness costs.

Without loss of generality, we assume the common processing time is unity. We refer to the time interval $(t - 1, t]$ as time period t . Let K be the total number of distinct product types that the plant manufactures and let L_k be the total number of customer orders of type k for $k = 1, \dots, K$. Define $u_{k,l}$ as the demand expressed in units of batches of the l th customer order of product type k with due date $d_{k,l}$ for $k = 1, \dots, K$ and $l = 1, \dots, L_k$. Note that the demand $u_{k,l}$ may not necessarily be an integer value. We assume the due dates are integer multiples of the processing time. Let $\alpha_{k,l}$ and $\beta_{k,l}$ be the per unit time earliness and tardiness cost, respectively, of the l th order of product type k for $k = 1, \dots, K$ and $l = 1, \dots, L_k$.

The total number of batches needed to satisfy all the demand of product type k , n_k , is

$$n_k = \left\lceil \sum_{l=1}^{L_k} u_{k,l} \right\rceil, \quad (1)$$

and the cumulative total number of batches across all products, N , is

$$N = \sum_{k=1}^K n_k. \quad (2)$$

Define the total number of customer orders, R , as

$$R = \sum_{k=1}^K L_k. \quad (3)$$

Define T as

$$T = N + \max_{k,l} (d_{k,l}). \quad (4)$$

For the N total batches and with both earliness and tardiness penalties, it can be shown that it can never be cheaper to schedule a batch later than time period T .

Recall, the decisions that need to be made are (1) the allocation of demand from different customers to the batches and (2) the time period in which a batch is scheduled. Hence, the decision variables are the fraction of a batch i used to satisfy the l th order of product type k , $x_{k,l,i}$, and $z_{k,i,t} = 1$, if period t produces the i th batch of type k for $k = 1, \dots, K$, $l = 1, \dots, L_k$, $i = 1, \dots, n_k$, and $t = 1, \dots, T$. The variables $x_{k,l,i}$ are associated with the batching decision. A particular batch can only satisfy the demand of customer orders of the same product type. Note that if all the demand is an integer multiple of the batch size (i.e., integer values for variables $u_{k,l}$), there is no batching and the only decision that needs to be made is scheduling. The variables $z_{k,i,t}$ are associated with the scheduling decision.

The cost of producing the l th order of type k in period t , $c_{k,l,t}$, is

$$c_{k,l,t} = \alpha_{k,l} \max(0, d_{k,l} - t) + \beta_{k,l} \max(0, t - d_{k,l}). \quad (5)$$

Refer to the following problem formulation as Batching/Scheduling,

$$\min \sum_{t=1}^T \sum_{k=1}^K \sum_{i=1}^{n_k} z_{k,i,t} \left(\sum_{l=1}^{L_k} c_{k,l,t} x_{k,l,i} \right) \quad (6)$$

Subject to:

4.1. Allocation of Orders to Batches

$$\sum_{l=1}^{L_k} x_{k,l,i} \leq 1 \quad \text{for } k = 1, \dots, K; i = 1, \dots, n_k;$$

4.2. Demand to Batch Allocation

$$\sum_{i=1}^{n_k} x_{k,l,i} = u_{k,l} \quad \text{for } k = 1, \dots, K; l = 1, \dots, L_k;$$

4.3. Batch Assignment

$$\sum_{t=1}^T z_{k,i,t} = 1 \quad \text{for } k = 1, \dots, K; i = 1, \dots, n_k;$$

4.4. Time Period Assignment

$$\sum_{k=1}^K \sum_{i=1}^{n_k} z_{k,i,t} \leq 1 \quad \text{for } t = 1, \dots, T;$$

4.5. Domain

$$x_{k,l,i} \geq 0, z_{k,i,t} \in \{0, 1\} \quad \text{for } k = 1, \dots, K; l = 1, \dots, L_k;$$

$$i = 1, \dots, n_k; t = 1, \dots, T.$$

The objective is to minimize the total weighted earliness and tardiness costs. Constraint 4.1 ensures that the total allocation to a batch is not more than the batch size. Constraint 4.2 allocates the demand from customer orders to batches to ensure that all demand is met. Constraint 4.3 assigns each batch to one production period. Constraint 4.4 ensures that not more than one batch is produced in each period.

Although we assume time-proportional earliness and tardiness penalties in the calculation of the cost variables, $c_{k,l,t}$, the above formulation is still valid for general penalties, since the cost variables can be any real number value. For these cases, the cost variables can be treated as a direct input into the formulation. However, for the purposes of this paper, we assume that they are calculated on the basis of time-proportional earliness and tardiness penalties.

Though our problem has its origin in a manufacturing domain, it is a special case of the capacitated-location problem (Francis, McGinnis, and White 1992). If we consider a batch as a location and the batch size as the capacity, then a batch can be thought of as a capacitated service location for the orders. The cost of service is the earliness or tardiness penalty associated with the order. Our specialization to the capacitated location problem has the fixed costs dominating the variable costs, all supplies equal to one, and the variable costs having a special cost structure associated with the earliness and tardiness penalties. Unfortunately, capacitated-location problems are known to be computationally hard. The best method known for these problems is branch-and-bound on their mixed integer formulation. We were not able to make use of any of the above special properties of our problem in order to reduce its computational complexity by finding a polynomial algorithm to optimally solve it.

Before we embark on the details of the procedure, we first discuss some other natural ways that can be tried to solve this problem. An easy-to-solve manufacturing problem that comes closest to our problem is the single-machine manufacturing problem with orders with integral demands, unit batch size, and identical processing time. This problem can be modeled as an assignment problem, which is well known to be an easy problem to solve. However, our problem has orders with nonintegral demands. In fact, the most difficult subset of our problem set are the problems where the demands are all significantly smaller than the batch size (which is assumed to be unity). As a result, a batching decision is required in addition to the scheduling decision. A natural heuristic would be to batch the customer demand in order of their due dates and then use the assignment formulation to solve the remaining problem. This is what we call a single-pass *serial* heuristic. It is included as a benchmark in the computational section.

In addition to the serial heuristic, we also have included as a benchmark a greedy heuristic, which finds the cheapest batches first. We have included this heuristic because often a greedy heuristic is found to be as good as more complicated schemes. For example, Bishop (1957) showed that a greedy-type rule determines an optimal single-product schedule when no tardiness is allowed. Dessouky and Kijowski (1997) extended this result for the multiproduct case. In our context where a batching decision is required as well as a scheduling decision and time-proportional tardiness costs are allowed, a greedy type rule is no longer optimal.

5. Heuristic Procedure

We present an iterative heuristic procedure that finds a local minimum to our problem. The iterative heuristic solves a series of linear programming relaxations of problem formulation Batching/Scheduling. The heuristic procedure makes use of the following two observations.

Observation 1. Given the periods in which the i th batch of type k , $k = 1, \dots, K$ and $i = 1, \dots, n_k$, is produced (batch schedule), the optimal allocation of customer orders to batches can be found by solving a transportation problem. We further note that there are K independent transportation problems, one for each product type.

Observation 2. Given the allocation of customer orders to batches, the optimal schedule of batches can be found by solving an assignment problem.

Observation 1 is easy to see, since problem formulation Batching/Scheduling reduces to a transportation problem if the batch schedule is fixed, that is, if the values of variables $z_{k,i,t}$, $k = 1, \dots, K$, $i = 1, \dots, n_k$, and $t = 1, \dots, T$ are known. The transportation problem is given by Constraints 4.1 and 4.2. Note that for this transportation problem, the solution (variables $x_{k,l,i}$) may not be integral, since the demand ($u_{k,l}$) may not necessarily be integral.

The transportation problem is a standard decision problem that can be solved using extremely efficient algorithms.

By fixing an allocation of demand to batches (variables $x_{k,l,i}$), the problem reduces to determining the assignment of the i th batch of type k to an available period within $[1, T]$ to minimize the total weighted earliness and tardiness costs for $k = 1, \dots, K$ and $i = 1, \dots, n_k$. The assignment problem is given by Constraints 4.3 and 4.4. The assignment problem is again a standard decision problem that can be solved efficiently.

We now present a heuristic, based on Observations 1 and 2, that iteratively solves a series of transportation and assignment problems. The procedure first starts with an initial batching (variables $x_{k,l,i}$) and solves the batch scheduling problem using the assignment problem to obtain a batch schedule (variables $z_{k,i,t}$). Then, given the batch schedule, the batching problem is solved using the transportation problem. The output of this step is a new set of batching variables $x_{k,l,i}$. This process is repeated until there is no further improvement in the objective value. The heuristic is summarized as follows.

1. *Initialization.* Set the current objective value w^{old} to a large number. Determine an initial batching strategy ($\bar{x}_{k,l,i}$ for $k = 1, \dots, K$, $l = 1, \dots, L_k$, and $i = 1, \dots, n_k$).

2. *Scheduling.* With the current batching ($\bar{x}_{k,l,i}$) solve the assignment problem to get the batch schedule and determine the new objective value w^{new} . Call the new batch schedule $\bar{z}_{k,i,t}$.

3. *Stopping Criteria.* If $w^{\text{new}} < w^{\text{old}}$, go to Step 4 to get a new batching strategy; otherwise, stop and the objective value is given by w^{new} .

4. *Batching.* With the current batch schedule ($\bar{z}_{k,i,t}$) solve the transportation problem to get the batching. Call the new batching $\bar{x}_{k,l,i}$. Go to Step 2.

The above procedure requires that an initial batching strategy be entered as input. One initial strategy is to group customer orders in the natural order of their due dates. Clearly, a different initial batching strategy may lead to a different local minimum. The procedure does not cycle, i.e., it terminates in a finite number of steps. This fact follows from the observation that at each step, a feasible solution is one of the finite set of extreme points of the polyhedron specified by the constraints of problem Batching/Scheduling. Since we terminate our heuristic if the objective function value does not decrease at a step, we cannot repeat an extreme point during the progress of the heuristic.

At Step 2 of the heuristic, we solve an assignment problem, where N batches are optimally assigned to N out of a total of T production periods. The complexity of the solution procedure to the assignment problem to get the batch schedule is $O(T^3)$ (Lawler 1976). However, we note that we need not consider all of the T production periods as possible busy periods. For example, if the due dates are scattered within $(0, T]$, we can identify some of the periods lying between the due dates that will be idle in an optimal solution. It can be easily shown that the number of busy periods that need to be considered is only $2N$. Hence, the complexity of the assignment problem reduces to $O(N^3)$. We note that it takes $O(R \log R)$ time to identify the set of candidate busy periods, where R is the total number of customer orders.

6. Special Cost Structure and Lower Bound

If the earliness penalties of customer orders of the same type are ranked in the same order of their due dates and tardiness penalties of customer orders of the same type are ranked in the reverse order of their due dates, we next show that the problem may be solved in $O(N^3 + R \log R)$, where N is the total number of batches to produce across all products and R is the total number of customer orders. We refer to this condition as the *Dominance Condition* and formally summarize it in the next definition. One obvious case that meets the dominance condition is when the customer orders of the same product type have the same penalty costs. A more practical example is the case in which orders of the same type have the same earliness penalty but the orders that are due earlier have a higher priority (e.g., higher tardiness

penalty). This is however expected to happen only if all the orders of one product type are from the same customer. Our primary reason to present this condition is to use it to derive a lower bound for the more general case in which this condition is not met.

Definition 1: Dominance Condition. The customer orders $1, 2, \dots, L_k$ for each product type k can be arranged such that $d_{k,1} \leq d_{k,2} \leq \dots \leq d_{k,L_k}$, $\alpha_{k,1} \leq \alpha_{k,2} \leq \dots \leq \alpha_{k,L_k}$, and $\beta_{k,1} \geq \beta_{k,2} \geq \dots \geq \beta_{k,L_k}$.

The only two reasons for not producing customer orders in the order of their indices could be that the tardiness cost of an order is more than that of an order with lower index or the earliness cost of an order is more than that of an order with higher index. Both the cases are impossible under the Dominance Condition. Therefore, we have the following property.

Property 1. Under the Dominance Condition, there exists an optimal batching and schedule such that for each product type, the customer orders are produced in order of their indices; that is, the $l + 1$ th customer order will be produced no sooner than the l th customer order.

This property implies an optimal batching for each product in order of the indices. Moreover, in case, $\sum_{l=1}^{L_k} u_{k,l}$ is integral, all the n_k batches are fully utilized to satisfy the customer orders. Therefore, we have the following.

COROLLARY 1. If the dominance condition holds, $\sum_{l=1}^{L_k} u_{k,l}$ is integral for each product type k and at Step 2 of the first iteration, ties in due dates are broken first in order of the earliness costs and next in the reverse order of the tardiness costs, then the heuristic finds the optimal batching and schedule in $O(N^3 + R \log R)$ time after Step 2 of the first iteration.

We now present a lower bound for problem Batching/Scheduling by transforming the earliness and tardiness penalties to satisfy the dominance condition. We find the new penalties such that for each order, they are the largest penalties less than or equal to the original penalties while satisfying the Dominance Condition. This is easily accomplished in linear complexity. If we further assume that $\sum_{l=1}^{L_k} u_{k,l}$ is integral for each product type k , then according to Corollary 1, the problem with the transformed penalties may be optimally solved by the heuristic. Since for each order the transformed earliness and tardiness penalties are smaller than the corresponding actual penalties, an optimal solution to the problem with the transformed penalties is a lower bound to the problem with the actual penalties.

7. Computational Experiments

We compare the proposed iterative heuristic against two heuristics referred to as greedy and serial. The greedy heuristic allocates and schedules one batch at a time. The orders that are *allocated* to the first scheduled batch are the ones that give the smallest cost. This process is repeated until all orders have been allocated to a batch and the resulting batch has been scheduled. The greedy heuristic represents the situation in which managers make sequential decisions on the batches and for the given batch the cheapest allocation and schedule is selected. This process contrasts with the proposed iterative heuristic, where the allocation and scheduling decisions are made collectively across all batches. The serial heuristic contrasts with the proposed heuristic by stopping after one iteration. It represents the case in which the batching (allocation) and scheduling decisions are made serially. In this case, the planning department determines the customer allocation to batches, and then another department determines the schedule for each batch. Comparing the serial heuristic with the iterative heuristic shows the significance of integrating the two decisions, batching and scheduling.

We also benchmark the proposed iterative heuristic against optimal solutions generated by CPLEX, a commercially available optimization software package. In cases where CPLEX is not able to find an optimal solution, we use the lower bound presented in Section 6 as the benchmark.

The optimization model formulation presented in Section 4 is a mixed-integer nonlinear programming model that cannot easily be solved using CPLEX. Hence, to obtain solutions

using CPLEX, the presented model is reformulated to a mixed-integer linear formulation. The following variable transformations are necessary.

$$y_{k,l,t} = \sum_i^{n_k} x_{k,l,t} z_{k,i,t} \quad \forall k, l, t \quad (7)$$

$$v_{k,t} = \sum_i^{n_k} z_{k,i,t} \quad \forall k, t \quad (8)$$

The nonlinear programming model is presented in Section 4 instead of the integer programming model because the heuristic procedures follow directly from the prior model.

We use the *smallest lower bound* as the node selection rule in CPLEX. The stopping rule used in CPLEX is when 10 MB of memory is reached. The experiments are run on an IBM RS/6000 Workstation.

Tested values for the total demand across all product types expressed in units of batches, Q , are 10, 20, 40, and 80. The earliness and tardiness penalties are generated from a continuous uniform random number from (0, 100]. The product type for each customer order is generated from a discrete uniform random number from (0, K_{\max}]. Tested values for K_{\max} are 2, 4, and 8. A high K_{\max} represents a scenario when the factory is producing a high product mix. The demand size per customer order is generated from a continuous uniform random number from (0, U_{\max}]. Tested values for U_{\max} are 1 and 5. A scenario with a small U_{\max} means that a particular batch is more likely to satisfy the demand of many small size customer orders and is representative of operating a large batch size. A large U_{\max} is for the case in which most of the batches satisfy the demand of a single customer order, with little mixing of orders in a batch, and is representative of a situation with a small batch size. Due dates for each customer order are generated from a discrete uniform random number from (0, D_{\max}]. Tested values for D_{\max} are Q and $2Q$. Scenarios with $D_{\max} = Q$ represent a tight due date case while the ones with $D_{\max} = 2Q$ represent a loose due date case.

Recall for the proposed pesticide plant, there was planned to be six mixing tank lines, each with its own dedicated set of products. The lowest product mix of any line was 2, while the highest was 9. Batch sizes that were being considered were 5,000 and 10,000 liters, and the different mixing tanks could have different batch sizes. As previously mentioned, in relation to the order sizes, using a higher batch size results in more batching and provides more slack in meeting the due dates. Hence, the low U_{\max} and high D_{\max} scenarios reflect the situation of a line, using a large batch size. In summary, the above experimental design covers the different spectrum of possible operating environments of the six different mixing tank lines for the proposed pesticide plant.

Table 1 lists the results of the experiments. For each scenario, 30 runs are performed. The values in the fifth column in Table 1 represent the average ratio of the proposed iterative heuristic solution to the greedy heuristic solution. The values in the sixth column represent the average ratio of the iterative heuristic to the serial heuristic solution. Compared to the iterative heuristic, the greedy heuristic provides poor results. Although the greedy heuristic ensures that the allocation and schedule of some of the batches are performed as cheaply as possible, the overall allocation and schedule is poor. The results also show that an iterative procedure is significantly better than a serial procedure when U_{\max} is small; that is, when the order sizes are smaller than the batch size, the batching decision becomes more important and it is significantly better to integrate the two problem domains, batching and scheduling.

We remark that for all scenarios, the three heuristics are computationally fast requiring at most one CPU minute in any given tested scenario. The iterative heuristic is computational efficient, since in all scenarios a solution is found within 10 iterations.

We previously showed that the iterative heuristic outperforms two other heuristics. We

TABLE 1
Summary of Results

Q	U_{\max}	K_{\max}	D_{\max}	Iterative/Greedy	Iterative/Serial	Optimal/Iterative	Lower/Iterative
10	1	2	1 Q	0.44	0.71	0.98	0.98
10	1	4	1 Q	0.43	0.76	0.96	0.96
10	1	8	1 Q	0.41	0.83	0.99	0.98
10	5	2	1 Q	0.64	0.98	1.00	1.00
10	5	4	1 Q	0.60	0.97	1.00	1.00
10	5	8	1 Q	0.56	0.98	1.00	1.00
10	1	2	2 Q	0.60	0.56	0.92	0.92
10	1	4	2 Q	0.59	0.52	0.87	0.89
10	1	8	2 Q	0.61	0.55	0.96	0.96
10	5	2	2 Q	0.76	0.94	0.99	0.99
10	5	4	2 Q	0.72	0.94	0.99	0.99
10	5	8	2 Q	0.75	0.96	0.99	0.99
20	1	2	1 Q	0.33	0.64	0.95	0.96
20	1	4	1 Q	0.37	0.68	0.97	0.89
20	1	8	1 Q	0.37	0.74	—	0.75
20	5	2	1 Q	0.56	0.93	0.99	0.99
20	5	4	1 Q	0.48	0.93	0.97	0.99
20	5	8	1 Q	0.52	0.95	0.97	0.96
20	1	2	2 Q	0.54	0.58	—	0.63
20	1	4	2 Q	0.54	0.49	—	0.67
20	1	8	2 Q	0.55	0.51	—	0.73
20	5	2	2 Q	0.70	0.83	0.98	0.97
20	5	4	2 Q	0.70	0.79	0.97	0.96
20	5	8	2 Q	0.73	0.86	0.97	0.96
40	1	2	1 Q	0.27	0.59	—	0.89
40	1	4	1 Q	0.27	0.61	—	0.66
40	1	8	1 Q	0.34	0.67	—	0.57
40	5	2	1 Q	0.45	0.93	0.99	0.99
40	5	4	1 Q	0.44	0.90	0.93	0.95
40	5	8	1 Q	0.45	0.93	—	0.91
40	1	2	2 Q	0.41	0.55	—	0.41
40	1	4	2 Q	0.46	0.55	—	0.46
40	1	8	2 Q	0.58	0.52	—	0.65
40	5	2	2 Q	0.64	0.83	—	0.87
40	5	4	2 Q	0.69	0.74	—	0.81
40	5	8	2 Q	0.68	0.76	—	0.86
80	1	2	1 Q	0.21	0.54	—	0.80
80	1	4	1 Q	0.24	0.57	—	0.59
80	1	8	1 Q	0.29	0.61	—	0.45
80	5	2	1 Q	0.36	0.90	—	0.97
80	5	4	1 Q	0.36	0.86	—	0.91
80	5	8	1 Q	0.38	0.89	—	0.84
80	1	2	2 Q	0.31	0.52	—	0.28
80	1	4	2 Q	0.37	0.52	—	0.34
80	1	8	2 Q	0.45	0.51	—	0.48
80	5	2	2 Q	0.57	0.80	—	0.82
80	5	4	2 Q	0.57	0.69	—	0.71
80	5	8	2 Q	0.61	0.68	—	0.71

now evaluate how closely it approximates the global minimum. The Optimal and Lower columns in Table 1 show this comparison. The Optimal column gives the average ratio of the global minimum over the iterative heuristic solution. In cases where CPLEX is not able to find an optimal solution based on the given stopping rule, we benchmark the heuristic solution against the better of either the lower bound presented in Section 6 or the one provided by CPLEX.

As Table 1 shows, the iterative heuristic solution provides a close approximation to the global minimum when either Q is small or when U_{\max} is large. The results also show for scenarios when the due dates are tight (small D_{\max}) and the product mix is low (small K_{\max}), the heuristic also provides a close approximation for the case of large Q and small U_{\max} . For scenarios yielding a small average ratio of lower bound over heuristic solution, no conclusion concerning the quality of the iterative heuristic solution as compared to the global minimum can be made, since either the lower bound or the heuristic solution may not be tight approximations on the optimum for these scenarios.

As a final remark, we also tested the sensitivity of the results to the earliness penalties. Typically, customer orders requesting the same product type will have the same earliness weight, since they have the same inventory holding cost ($\alpha_{k,l} = \alpha_k$ for $l = 1, \dots, L_k$). We found that the same general findings as before still hold for this situation.

8. Conclusions and Future Research

In this paper, we introduce the integrated batching and scheduling problem in context of batch chemical processing with fixed batch sizes, sequence-independent setup times, and identical processing times. The joint batching and scheduling problem is formulated as a mixed-integer nonlinear programming model. An iterative heuristic procedure that solves a series of transportation and assignment problems is proposed. The heuristic terminates when a local minimum to the problem is found. In an experimental section we show that the iterative heuristic outperforms a greedy heuristic and a serial heuristic, which treats the two problem domains separately. Furthermore, we show that the local minimum found by the iterative heuristic is a close approximation to the global minimum, when the total demand is small (small Q) or when the order sizes are large compared to the batch size (large U_{\max}).

The presented heuristic can be in principal adapted for more complex problems. We have shown the possibility that certain integration problems might be solved reasonably well by first breaking the problem into smaller components that can be easily solved and then iteratively improve the solution by making several passes at the individual components. For example, in this paper, we assumed identical processing times for all product types. Even if we relax that assumption, we can still use a similar approach of breaking down the problem into batching and scheduling components. Only now, the scheduling component is further broken into as many components as there are different processing times. The whole process can be further improved by adding more starting points at its first step. Subjects of future research are to generalize this technique in a formal way and to find quantitative bounds for the quality of solutions achieved in this manner.

References

- BAKER, K. R. AND G. D. SCUDDER (1990), "Sequencing With Earliness and Tardiness Penalties: A Review," *Operations Research*, 38, 1, 22–36.
- BIREWAR, B. D. AND I. E. GROSSMAN (1990), "Simultaneous Production Planning and Scheduling in Multiproduct Batch Plants," *Industrial and Chemical Engineering Research*, 29, 4, 570–580.
- BISHOP, G. T. (1957), "On a Problem of Production Scheduling," *Operations Research*, 5, 1, 97–103.
- CPLEX Optimization (1996), 930 Tahoe Boulevard, Suite 802-279, Incline Village, NV 89451.
- CHAUDHARY, J. (1988), "Batch Plants Adapt to CPI's Flexible Game Plans," *Chemical Engineering*, 95, 2, 31–36.
- DESSOUKY, Y. M., C. A. ROBERTS, M. M. DESSOUKY, AND G. WILSON (1996), "Scheduling Multi-Purpose Batch Plants with Junction Constraints," *International Journal of Production Research*, 34, 2, 525–541.
- DESSOUKY, M. M. AND B. A. KIJOWSKI (1997), "Production Scheduling of Single-Stage Multi-Product Batch Chemical Processes with Fixed Batch Sizes," *IIE Transactions*, 29, 5, 399–408.
- FAQIR, N. M. AND I. KARIMI (1989), "Optimal Design of Batch Plants with Single Production Routes," *Industrial and Chemical Engineering Research*, 28, 5, 1191–1202.
- FRANCIS, R. L., L. F. MCGINNIS, AND J. A. WHITE (1992), *Facility Layout and Location: An Analytical Approach*, Prentice Hall, Englewood Cliffs, NJ.

- HEUTS, R. M. J., H. P. SEIDEL, AND W. J. SELEN (1992), "A Comparison of Two Lot Sizing-Sequencing Heuristics for the Process Industry," *European Journal of Operational Research*, 59, 3, 413–424.
- KONDILI, E., C. C. PANTELIDES, AND R. W. H. SARGENT (1993), "A General Algorithm for Short-term Scheduling of Batch Operations-I. MILP Formulation," *Computers and Chemical Engineering*, 17, 2, 211–227.
- KU, H. AND I. KARIMI (1990), "Scheduling in Serial Multiproduct Batch Processes with Due-Date Penalties," *Industrial and Chemical Engineering Research*, 29, 4, 580–590.
- KU, H., D. RAJAGOPALAN, AND I. KARIMI (1987), "Scheduling in Batch Processes," *Chemical Engineering Progress*, 83, 8, 35–45.
- KUDVA, G., A. ELKAMEL, J. F. PEKNY, AND G. V. REKLAITIS (1994), "Heuristic Algorithm for Scheduling Batch and Semi-Continuous Plants with Production Deadlines, Intermediate Storage Limitations and Equipment Changeover Costs," *Computers and Chemical Engineering*, 18, 9, 859–875.
- LAWLER, E. L. (1976), *Combinatorial Optimization: Networks and Matroids*. Holt, Rinehart, and Winston, New York.
- PATSIDOU, E. P. AND J. C. KANTOR (1991), "Scheduling of a Multipurpose Batch Plant Using a Graphically Derive Mixed-Integer Linear Program," *Industrial and Chemical Engineering Research*, 30, 7, 1548–1561.
- PEKNY, J. F., D. L. MILLER, AND G. J. MCRAE (1990), "An Exact Parallel Algorithm for Scheduling when Production Costs Depend on Consecutive System States," *Computers and Chemical Engineering*, 14, 7, 1009–1023.
- POTTS, C. N. AND L. N. VAN WASSENHOVE (1992), "Integrating Scheduling with Batching and Lot-Sizing: A Review of Algorithms and Complexity," *Journal of Operational Research Society*, 43, 5, 395–406.
- REKLAITIS, G. V. (1991), "Perspectives on Scheduling and Planning of Process Operations," *Proceedings Fourth International Symposium of Process Systems Engineering*, Montebello, Canada.
- RIPPIN, D. W. (1991), "The Big Advantage Offered by Batch Processing is Flexibility," *Chemical Engineering*, 98, 5, 101–107.
- SELEN, W. J. AND R. M. J. HEUTS (1990), "Operational Production Planning in a Chemical Manufacturing Environment," *European Journal of Operational Research*, 45, 1, 38–46.
- SMITH-DANIELS, V. L. AND L. P. RITZMAN (1988), "A Model for Lot Sizing and Sequencing in Process Industries," *International Journal of Production Research*, 26, 4, 647–674.

Maged M. Dessouky is an Associate Professor of Industrial and Systems Engineering at the University of Southern California. He received his Ph.D. in Industrial Engineering from the University of California, Berkeley, and M.S.I.E. and B.S.I.E. from Purdue University. His research interests include production and operations management, modeling of manufacturing processes and systems, simulation, and operations research applications to industrial systems. He has published in refereed journals such as *Mathematics of Operations Research*, *IIE Transactions*, *European Journal of Operational Research*, and *International Journal of Production Research*, among others.

Brian A. Kijowski has worked for General Motors and consulted for NASA, Reynolds Metals, and the Army Construction Engineering Research Laboratory. His research interests include production and operations management, applied mathematical programming, and the design and analysis of algorithms. He became interested in batch chemical scheduling while he was a visiting assistant professor at the University of Southern California. He currently is an independent consultant specializing in project management.

Sushil Verma received a B.Tech. degree in mechanical engineering from Indian Institute of Technology at New Delhi in 1987. He finished his Ph.D. in operations research from University of California at Berkeley in 1994. Soon after graduation, he joined the faculty of University of Southern California as a visiting assistant professor, where he was involved in an active research program as well as undergraduate teaching. He has published in refereed journals such as *Mathematical Programming* and *Mathematics of Operations Research*. He is currently working as a Consultant for i2 Technologies in Redwood City, CA.

POM-2000 at San Antonio
April 1–4, 2000

POM-2001 at Orlando
March 30–April 2, 2001