PROCESS DESIGN AND CONTROL

Robust Optimization for Process Scheduling Under Uncertainty

Zukui Li and Marianthi G. Ierapetritou*

Department of Chemical and Biochemical Engineering, Rutgers University, Piscataway, New Jersey 08854

This paper addresses the uncertainty problem in process scheduling using robust optimization. Compared to the traditional-scenario-based stochastic programming method, robust counterpart optimization method has a unique advantage, in that the scale of the corresponding optimization problem does not increase exponentially with the number of the uncertain parameters. Three robust counterpart optimization formulations—including Soyster's worst-case scenario formulation, Ben-Tal and Nemirovski's formulation, and a formulation proposed by Bertsimas and Sim—are studied and applied to uncertain scheduling problems in this paper. The results show that the formulation proposed by Bertsimas and Sim is the most appropriate model for uncertain scheduling problems, because it has the following advantages: (i) the model has the same size as the other formulations, (ii) it preserves its linearity, and (iii) it has the ability to control the degree of conservatism for every constraint and guarantees feasibility for the robust optimization problem.

1. Introduction

Uncertainty is a very important concern in real plants for process scheduling, because many of the parameters associated with scheduling are not known exactly. Parameters such as raw material availability, prices, machine reliability, processing, or duration time and market requirements vary with respect to time and are often subject to unexpected deviations, which can cause infeasibilities and production disturbances. Thus, scheduling under uncertainty recently has received much attention in the open literature from chemical engineering and operations research communities.

According to the different treatment of uncertainty, scheduling methods can be divided into two groups: reactive scheduling and preventive scheduling. Reactive scheduling involves the problem of modifying the original scheduling policy or generating scheduling policy on time when uncertainty occurs. On the other hand, with preventive scheduling, the objective is to generate robust scheduling policies before the uncertainty occurs. Almost all techniques that involve uncertainty try to find solutions that are flexible to changes to the input data. Although this solution might not be optimal, the target is to be as close as possible to the optimal one. Robust scheduling focuses on obtaining preventive schedules that minimize the effects of disruptions on the performance measure, and it tries to ensure that the preventive schedules maintain a high level of performance.

For the problem of robust schedule generation, different methods have been proposed in the literature. Generally, the formulations can be classified into two groups: (a) scenario-based stochastic programming formulation and (b) robust counterpart optimization formulation.

In the literature, most existing work on robust scheduling has followed the scenario-based formulation. Mulvey et al.[1] developed the scenario-based robust optimization to handle the tradeoff associated with solution and model robustness. A solution to an optimization is considered to be "solution robust" if it remains close to the optimal for all scenarios and "model robust" if it remains feasible for most scenarios. Kouvelis et al.[2] made the first attempts to introduce the concept of robustness for scheduling problems. They suggest a robust schedule when processing times are uncertain and compute a robust schedule based on the maximum absolute deviation between the robust solution and all the possible scenarios; however, this requires knowledge of all possible scenarios. Moreover, the optimal solution of each scenario is supposed to be known a priori. Vin and Ierapetritou[3] addressed the problem of quantifying the schedule robustness under demand uncertainty, introduced several metrics to evaluate the robustness of a schedule, and proposed a multiperiod programming model using extreme points of the demand range as scenarios to improve the schedule performance of batch plants under demand uncertainty. Using flexibility analysis, they observed that the schedules from the multiperiod programming approach were more robust than the deterministic schedules. Balasubramanian and Grossmann[4] proposed a multiperiod mixed-integer linear programming (MILP) model for scheduling multistage flowshop plants with uncertain processing times. They minimized the expected makespan and developed a special branch-and-bound algorithm with an aggregated probability model. The scenario-based approaches provide a straightforward way to implicitly incorporate uncertainty. However, they inevitably enlarge the size of the problem significantly as the number of scenarios increases exponentially with the number of uncertain parameters. This main drawback limits the application of these approaches to solve practical problems with a large number of uncertain parameters. Jia and Ierapetritou[5] proposed a multiobjective robust optimization model to address the problem of uncertainty in scheduling, considering the expected performance (makespan), model robustness, and solution robustness. The Normal Boundary Intersection (NBI) technique is utilized to solve the multiobjective model and successfully produce a Pareto optimal surface that captures the tradeoff among different objectives in the face of uncertainty. The schedules obtained by solving this multiobjective optimization problem include robust assignments that can accommodate demand uncertainty.

Although those uncertainty-handling frameworks that follow the scenario-based formulation target the generation of robust solutions, with respect to optimality and feasibility, the model

* To whom correspondence should be addressed. Tel.: 732-445-2971. Fax: 732-445-2421. E-mail: marianth@soemail.rutgers.edu.

10.1021/ie071431u CCC: $40.75 © 2008 American Chemical Society
Published on Web 05/24/2008

still has the common drawback of scenario-based formulations: it requires some statistic knowledge of the input data, which, in many cases, may be difficult to acquire. Moreover, optimization of expectations is a practice of questionable validity in processes that involve only a small number of "trials", because the benefits of an optimum expected value can only be visible in the long term of a large number of trials. Finally, for the scenario-based robust optimization method, the uncertainty is modeled through the use of several scenarios. This type of method provides a direct way to incorporate uncertainty. However, the problem size will increase exponentially with the number of uncertain parameters, which restricts its application in solving problems with many uncertain parameters.

As an alternative to the scenario-based formulation, the robust counterpart optimization has been proposed, which avoids the shortcomings of the scenario-based formulation. The underlying framework of the robust counterpart scheduling formulation is based on solving the robust counterpart optimization problem for the uncertain scheduling problem.

One of the earliest papers on robust counterpart optimization, by Soyster,[6] considered simple perturbations in the data, for the purpose of determining a reformulation of the original problem such that the resulting solution would be feasible under all possible perturbations. The pioneering work by Ben-Tal and Nemirovski,[7] El-Ghaoui et al.,[8] and Bertsimas and Sim[9] extended the framework of robust counterpart optimization, and that work included sophisticated solution techniques with nontrivial uncertainty sets that described the data. The major advantages of robust counterpart optimization, compared to scenario-based stochastic programming, are that (i) no assumptions are needed regarding the underlying probability distribution of the uncertain data and (ii) it provides a way of incorporating different attitudes toward risk.

For the problem of process scheduling under uncertainty, only very few works have been reported in robust counterpart optimization for generating robust schedules. Lin et al.[10] proposed a robust optimization method to address the problem of scheduling with uncertain processing times, market demands, or prices. The robust optimization model was derived from its deterministic model by considering the worst-case values of the uncertain parameters, when uncertainty is in a bounded form. They also studied the case that uncertainty is described by a known probability distribution, where the robust optimization formulation introduces a small number of auxiliary variables and additional constraints into the original MILP problem, generating a deterministic robust counterpart problem, which provides the optimal/feasible solution, given the (relative) magnitude of the uncertain data, a feasibility tolerance, and a reliability level.

In this paper, we compare several robust counterpart optimization formulations and proposed the appropriate formulation for scheduling under uncertainty. The remainder of this paper is organized as follows. Section 2 introduces the robust counterpart optimization formulations and their extensions to mixed integer linear programming problems. In section 3, the problem of scheduling under uncertainty is studied based on robust optimization considering specific uncertainties. Several case studies are presented in section 4, to illustrate the application of different robust formulations and present some comparison results; section 5 summarizes the main conclusions of the paper.

2. Robust Optimization

Compared to scenario-based stochastic programming, robust counterpart optimization represents a more systematic approach for optimization under uncertainty to determine flexible solutions. The objective of robust counterpart optimization is to choose a solution that is able to cope best with the various realizations of the uncertain data. The uncertainty data are assumed to be unknown but bounded, and most current research assumes convexity of the uncertainty space.

The optimization problem with uncertainty parameters is reformulated into a robust counterpart optimization problem. Unlike stochastic programming, robust optimization does not require information about the probability distribution of the uncertainty data, and it does not optimize an expected value objective function. Robust optimization promises to essentially ensure robustness and flexibility by enforcing the feasibility of an optimization problem for the entire given uncertainty space.

In this section, three robust counterpart optimization formulations are presented, assuming the following general MILP problem:

$$\begin{aligned} &\max\ cx \\ &\text{s.t.}\quad \sum_m a_{lm} x_m \le p_l\ \forall\ l \\ &x_m \text{ binary or continuous,}\quad m = 1, 2..., n \end{aligned} \tag{P1}$$

For the remainder of the paper, we assume, without any loss of generality, that the data uncertainty affects only the elements of the left-hand-side (LHS) matrix coefficients, for the following reasons:

(1) The objective function can be transformed to a constraint;

(2) If the right-hand-side (RHS) constant p_l is subject to uncertainty, we can introduce a new variable x_{n+1}, which is a binary variable with a fixed value of 1, and the original constraint is transformed to the following:

$$\sum_m a_{lm} x_m - p_l x_{n+1} \le 0 \tag{1}$$

$$1 \le x_{n+1} \le 1 \tag{2}$$

Assuming that a_{lm} are uncertain parameters, the constraints in expression P1 are expanded as follows:

$$\sum_{m \notin M_l} a_{lm} x_m + \sum_{m \in M_l} \tilde{a}_{lm} x_m \le p_l \tag{3}$$

where M_l denotes the index set for the uncertain coefficients in the lth constraint; $\tilde{a}_{lm}$ represents the true values for the coefficient parameters that take a value within the uncertainty range $[a_{lm} - \hat{a}_{lm}, a_{lm} + \hat{a}_{lm}]$, where$a_{lm}$ represents nominal values and $\hat{a}_{lm}$ represents the variation amplitude.

2.1. Soyster's Formulation. Robust counterpart optimization can be traced back to the work of Soyster,[6] which is the first work that considered coefficient uncertainty in linear programming formulations and showed that such uncertainty can be handled by an equivalent linear programming model. However, the approach admits the highest protection and is the most conservative one, because it ensures feasibility against all potential realizations. Thus, it corresponds to the worst-case version of the scenario approach.

The worst-case formulation of eq 3 then take the following form, considering all uncertain parameters to take their boundary values, the intention of which is to ensure that, for every possible value of the uncertain coefficient, the solution remains feasible:

$$\sum_{m\notin M_l} a_{lm}x_m + \sum_{m\in M_l} a_{lm}x_m + \sum_{m\in M_l} \hat{a}_{lm}|x_m| \le p_l \tag{4}$$

To eliminate the absolute value, an auxiliary variable (u_m) is introduced for eachx_m, $m \in M_l$, which defines new bounds for x_m. Thus, the following robust formulation (expression P2) is obtained:

$$\begin{aligned} &\max\ cx \\ &\text{s.t.}\ \sum_{m\notin M_l} a_{lm}x_m + \sum_{m\in M_l} a_{lm}x_m + \sum_{m\in M_l} \hat{a}_{lm}u_m \le p_l \\ &\begin{cases} u_m = x_m & (\text{if } x_m \text{ is positive or a binary variable}) \\ u_m \le x_m \le u_m & (\text{otherwise}) \end{cases} \end{aligned} \tag{P2}$$

In this formulation (expression P2), if the uncertainty parameter coefficient is positive or a binary variable, then no auxiliary variable and constraint will be added, because the absolute value is eliminated naturally. The robust formulation (ε,σ)-Interval Robust Counterpart (IRC[ε,σ]) proposed by Lin et al.[10] belongs to this type of formulation. The difference is that they add certain infeasibility tolerance to the constraints, to increase the level of control toward conservative solutions. The motivation to use Soyster's formulation is to provide maximum protection against uncertainty. This type of formulation allows for mitigation of the worse-case scenario; however, because all possible realizations of the data are considered, the solution can end up being overly pessimistic and the problem is more likely to be infeasible. For example, a worst-case parameter combination where all the processing times and all the demands take the maximum value might lead to an infeasible schedule in a fixed time horizon, because of inability of the plant to satisfy the demand in a fixed time horizon.

2.2. Ben-Tal and Nemirovski's Formulation. Since Soyster's formulation is extremely conservative, it is highly desirable to provide a mechanism to allow a tradeoff between robustness and performance. A significant step forward for developing a theory for robust optimization was taken by Ben-Tal and Nemirovski,[11] who proposed the following robust counterpart formulation:

$$\begin{aligned} &\max\ cx \\ &\sum_{m} a_{lm}x_m + \sum_{m\in M_l} \hat{a}_{lm}u_{lm} + \Omega_l\sqrt{\sum_{m\in M_l}\hat{a}_{lm}^2 z_{lm}^2} \le p_l \\ &\qquad - u_{lm} \le x_m - z_{lm} \le u_{lm} \end{aligned} \tag{P3}$$

Let us consider that the values of the uncertainty coefficients are obtained through random perturbations:

$$\tilde{a}_{lm} = a_{lm} + \xi_{lm}\hat{a}_{lm} \tag{5}$$

where $\{\xi_{lm}\}$($m \in M_l$) are independent random variables that are symmetrically distributed in interval [−1, 1]. As shown by Ben-Tal and Nemirovski,[11] this robust formulation ensures that the probability that the lth constraint is violated is, at most, $e^{-\Omega_l^2}/2$, i.e.,

$$\Pr\left\{\sum_{m\notin M_l} a_{lm}x_m + \sum_{m\in M_l}\tilde{a}_{lm}x_m \ge p_l\right\} \le \kappa_l \tag{6}$$

$$\kappa_l = e^{-\Omega_l^2/2} \tag{7}$$

This robust optimization formulation was first introduced for linear programming problems with uncertain linear coefficients and is extended by Lin et al.[10] to MILP problems under uncertainty. The robust formulation (ε,σ,κ)-Robust Counterpart (RC[ε,σ,κ]) proposed by the authors belongs to this type of formulation. This type of robust counterpart formulation has the flexibility of controlling the degree of solution conservatism through the constraint violation probability $e^{-\Omega_l^2}/2$. Its main drawback is that it corresponds to a nonlinear optimization formulation.

2.3. Bertsimas and Sim's Formulation. Although Ben-Tal and Nemirovski's robust formulation provides a way to consider the tradeoff between performance and robustness, it results in a nonlinear formulation. To avoid the complication of a nonlinear optimization, Bertsimas and Sim[9] considers robust linear programming with coefficient uncertainty, using an uncertainty set with budgets. In this robust counterpart optimization formulation, a budget parameter Γ_l (which takes a value between zero and the number of uncertain coefficient parameters in the constraints and is not necessarily an integer) is introduced to control the degree of conservatism of the solution. In other words, it is unlikely that all of the uncertain coefficient parameters will get the worst-case value at the same time, so the goal of this formulation is to control that up to $\lfloor\Gamma_l\rfloor$ of those parameters are allowed to get their worst-case value.

$$\sum_m a_{lm}x_m + \max_{\{S_l\cup\{t_l\}|S_l\subseteq M_l,|S_l|=\lfloor\Gamma_l\rfloor,t_l\in M_l\backslash S_l\}}\left\{\sum_{m\in S_l}\hat{a}_{lm}|x_m| + (\Gamma_l - \lfloor\Gamma_l\rfloor)\hat{a}_{lt_l}|x_{t_l}|\right\} \le p_l \tag{8}$$

where S_l represents the subset that contains $\lfloor\Gamma_l\rfloor$ uncertain parameters in the constraint, and t_l is an index to describe an additional uncertain parameter if Γ_l is not an integer. Thus, constraint 8 expresses the requirement that, up to $\lfloor\Gamma_l\rfloor$, uncertain parameters can get their worst-case values simultaneously, which can be clearly seen when Γ_l is chosen as an integer ($\Gamma_l = \lfloor\Gamma_l\rfloor$); the constraint given as eq 8 then becomes

$$\sum_m a_{lm}x_m + \max_{\{S_l|S_l\subseteq M_l,|S_l|=\Gamma_l\}}\left\{\sum_{m\in S_l}\hat{a}_{lm}|x_m|\right\} \le p_l \tag{9}$$

When Γ_l is not integer, one more uncertain parameter, a_{lt}, can change by ($\Gamma_l - \lfloor\Gamma_l\rfloor$)$\hat{a}_{lt_l}$. Thus, the robust formulation takes the following form:

$$\begin{aligned} &\max\ cx \\ &\text{s.t.}\ \sum_m a_{lm}x_m + \max_{\{S_l\cup\{t_l\}|S_l\subseteq M_l,|S_l|=\lfloor\Gamma_l\rfloor,t_l\in M_l\backslash S_l\}}\left\{\sum_{m\in S_l}\hat{a}_{lm}u_m + (\Gamma_l - \lfloor\Gamma_l\rfloor)\hat{a}_{lt_l}u_{t_l}\right\} \le p_l \\ &\begin{cases} u_m = x_m & (\text{if } x_m \text{ is positive or a binary variable}) \\ -u_m \le x_m \le u_m & (\text{otherwise}) \end{cases} \end{aligned} \tag{P4}$$

To transform problem P4 to a single optimization problem, let

$$\beta(x,\Gamma_l) = \max_{\{S_l\cup\{t_l\}|S_l\subseteq M_l,|S_l|=\lfloor\Gamma_l\rfloor,t_l\in M_l\backslash S_l\}}\left\{\sum_{m\in S_l}\hat{a}_{lm}u_m + (\Gamma_l - \lfloor\Gamma_l\rfloor)\hat{a}_{lt_l}u_{t_l}\right\} \tag{10}$$

Then, $\beta(x,\Gamma_l)$ is equal to the objective of optimization problem P5, because the optimal solution z_{lm}^* of problem P5 must consist of $\lfloor\Gamma_l\rfloor$ variables at 1 and one variable at $\Gamma_l - \lfloor\Gamma_l\rfloor$ with $u_m \ge 0$.

$$\begin{aligned} &\max \sum_{m\in M_l}\hat{a}_{lm}u_m z_{lm} \\ &\text{s.t.} \sum_m z_{lm} \le \Gamma_l \\ &0 \le z_{lm} \le 1 \end{aligned} \tag{P5}$$

The dual problem of problem P5 is described as follows:

$$\begin{aligned} \min \quad & \Gamma_l z_l + \sum_{m \in M_l} q_{lm} \\ \text{s.t.} \quad & z_l + q_{lm} \geq \hat{a}_{lm} u_m \\ & q_{lm} \geq 0 \\ & z_l \geq 0 \end{aligned} \qquad \text{(P6)}$$

where q_{lm} corresponds to the dual variable of the inequality $z_{lm} \leq 1$, z_l corresponds to the dual variable of the inequality $\sum_m z_{lm} \leq \Gamma_l$.

The robust formulation is transformed to the following equivalent formulation after substituting the inner optimization problem in expression P4 with the equivalent optimization problem (expression P6).

$$\begin{aligned} & \max \ cx \\ & \text{s .t.} \sum_m a_{lm} x_m + \Gamma_l z_l + \sum_{m \in M_l} q_{lm} \leq p_l \\ & \qquad z_l + q_{lm} \geq \hat{a}_{lm} u_m \\ & \qquad q_{lm} \geq 0 \\ & \qquad z_l \geq 0 \\ & \begin{cases} u_m = x_m & \text{(if } x_m \text{ is positive or a binary variable)} \\ -u_m \leq x_m \leq u_m & \text{(otherwise)} \end{cases} \end{aligned} \qquad \text{(P7)}$$

In this model, a budget parameter for each constraint in problem P1 Γ_l limits the number of coefficients that can simultaneously take their worst-case value; the resulting robust optimization remains a linear formulation. This is different from the worst-case formulation, where all the parameters are considered to get their worst-case values at the same time, without control of conservatism of the solution. Also note that formulation P7 maintains its linearity. For this robust counterpart formulation, Bertsimas and Sim[9] calculated the probability bounds of the constraint violation. Specifically, if the uncertain coefficient parameter $\tilde{a}_{ij}$ follows a symmetric distribution and takes values in the range $[a_{ij} - \hat{a}_{ij}, a_{ij} + \hat{a}_{ij}]$, then the probability that the ith constraint is violated satisfies the following constraint:

$$P\left(\sum_m \tilde{a}_{lm} x_m > p_l\right) \leq \frac{1}{2^n}\left\{(1-\mu)\sum_{l=\lfloor v \rfloor}^{n} \binom{n}{l} + \mu \sum_{l=\lfloor v \rfloor+1}^{n} \binom{n}{l}\right\} \leq (1-\mu)C(n,\lfloor v \rfloor) + \sum_{k=\lfloor v \rfloor+1}^{n} C(n,k) \qquad (11)$$

where

$$n = |M_l|, v = \frac{\Gamma_l + n}{2}, \mu = v - \lfloor v \rfloor$$

$$C(n,k) = \begin{cases} \dfrac{1}{2^n} & \text{(if } k = 0 \ \text{ or } k = n) \\ \dfrac{1}{\sqrt{2\pi}}\sqrt{\dfrac{n}{(n-k)k}} \exp\left\{n \log\left[\dfrac{n}{2(n-k)}\right] + k \log\left(\dfrac{n-k}{k}\right)\right\} & \text{(otherwise)} \end{cases}$$

When applying this robust counterpart formulation to practical problems with a large number of uncertain parameters, we can use the feasibility test to identify a priori which parameters are allowed to take the worst-case value, thus providing a guide of assigning appropriate budget parameters to the different constraints.

2.4. Comparison of Different Formulations. For a deterministic MILP problem with n variables, and m constraints (not counting the bounding constraints) having a total of k uncertain parameters, where a number j of all the constraints are subject to uncertainty and a number q of all the decision variables are subject to uncertain coefficients, we have the following comparison for the three different robust formulations presented in the previous section, in terms of the number of variables, the number of constraints required, the type of formulation, and the information obtained:

(1) Formulation 1, by Soyster:[6] this has $n + q$ variables and $m + 2q$ constraints, and it is a linear formulation; however, it provides no control over the degree of conservatism of the solution.

(2) Formulation 2, by Ben-Tal and Nemirovski:[11] this is a second-order cone problem (nonlinear) that has $n + 2k$ variables and $m + 2k$ constraints, and it is able to control the degree of conservatism through the constraint violation probability parameter.

(3) Formulation 3, by Bertsimas and Sim:[9] this is a linear optimization problem that has $n + j + k + q$ variables and $m + k + 2q$ constraints, and it also provides a way to control the degree of solution conservatism through the budget parameter Γ_l.

In summary, Soyster's worst-case formation is the simplest formulation, with the smallest number of variables and constraints, but it is not able to adjust the solution conservatism; thus, the generated solution often will be too pessimistic. Ben-Tal and Nemirovski's formulation provides a level of control for solution conservatism, but it results in a nonlinear formulation, which will cause computational complexity in solving mixed-integer nonlinear programming (MINLP) problems. On the other hand, the robust counterpart optimization formulation proposed by Bertsimas and Sim is a linear formulation that has the flexibility of adjusting the solution conservatism and also does not result in a substantial increase in problem size. Note that all the robust counterpart formulations have the same number of binary variables as the original deterministic formulation. In the next section, different uncertainties (price, processing time, and demand) in the process scheduling problem will be considered using Bertsimas and Sim's robust counterpart formulation, whereas a comparison of the three robust formulations will be conducted through different examples in section 4.

3. Robust Scheduling

For the general process scheduling problem, the following deterministic formulation (problem P8), which was proposed by Ierapetritou and Floudas,[12] is used:

$$\max \sum_{s \in S_p, n} \text{price}_s d_{s,n} - \sum_{s \in S_r} \text{price}_s(\text{STI}_s - \text{STF}_s) \qquad \text{(P8)}$$

$$\text{s.t.} \sum_{i \in I_j} wv_{i,j,n} \leq 1 \quad \forall\, i \in I \qquad (12)$$

$$\text{st}_{s,n} = \text{st}_{s,n-1} - d_{s,n} - \sum_{i \in I_s} \rho^{\text{P}}_{s,i} \sum_{j \in J_i} b_{i,j,n} + \sum_{i \in I_s} \rho^{\text{c}}_{s,i} \sum_{j \in J_i} b_{i,j,n-1} \quad \forall\, s \in S, \forall\, n \in N \qquad (13)$$

$$\text{st}_{s,n} \leq \text{st}^{\max}_s \quad \forall\, s \in S, \forall\, n \in N \qquad (14)$$

$$v^{\min}_{i,j} \text{wv}_{i,j,n} \leq b_{i,j,n} \leq v^{\max}_{i,j} \text{wv}_{i,j,n} \quad \forall\, i \in I, \forall\, j \in J_i, \forall\, n \in N \qquad (15)$$

$$\sum_{n} d_{s,n} \ge r_s \quad \forall s \in S \tag{16}$$

$$\mathrm{Tf}_{i,j,n} = \mathrm{Ts}_{i,j,n} + \alpha_{i,j}\mathrm{wv}_{i,j,n} + \beta_{i,j}b_{i,j,n} \quad \forall i \in I, \forall j \in J_i, \forall n \in N \tag{17}$$

$$\mathrm{Ts}_{i,j,n+1} \ge \mathrm{Tf}_{i,j,n} - H(1 - \mathrm{wv}_{i,j,n}) \quad \forall i \in I, \forall j \in J_i, \forall n \in N \tag{18}$$

$$\mathrm{Ts}_{i,j,n+1} \ge \mathrm{Tf}_{i',j,n} - H(1 - \mathrm{wv}_{i',j,n}) \quad \forall i, i' \in I_j, \forall j \in J, \forall n \in N \tag{19}$$

$$\mathrm{Ts}_{i,j,n+1} \ge \mathrm{Tf}_{i',j',n} - H(1 - \mathrm{wv}_{i',j',n}) \quad \forall i, i' \in I_j, i \ne i', \forall j, j' \in J, \forall n \in N \tag{20}$$

$$\mathrm{Ts}_{i,j,n+1} \ge \mathrm{Ts}_{i,j,n} \quad \forall i \in I, \forall j \in J_i, \forall n \in N \tag{21}$$

$$\mathrm{Tf}_{i,j,n+1} \ge \mathrm{Tf}_{i,j,n} \quad \forall i \in I, \forall j \in J_i, \forall n \in N \tag{22}$$

$$\mathrm{Ts}_{i,j,n} \le H \quad \forall i \in I, \forall j \in J_i, \forall n \in N \tag{23}$$

$$\mathrm{Tf}_{i,j,n} \le H \quad \forall i \in I, \forall j \in J_i, \forall n \in N \tag{24}$$

In the aforementioned formulation, the objective function is the profit (different performance measures can be used, such as makespan); the allocation constraints given by expression 12 state that only one of the tasks can be performed in each unit at an event point n; the constraints given by expression 13 represent the material balances for each state s, expressing that, at each event point n, the amount $\mathrm{st}_{s,n}$ is equal to that at event point $n - 1$, adjusted by any amount produced and consumed between event points $n - 1$ and n, and delivered to the market at event point n; the storage and capacity limitations of production units are expressed by the constraints given in expressions 14 and 15; the constraints given in expression 16 are written to satisfy the demands of the final products; and the constraints given in expressions 17–24 represent time limitations due to task duration and sequence requirements in the same or different production units. Although the aforementioned scheduling formulation has been used, the proposed methodology in this paper is not limited to this model, and it can also be applied to other scheduling formulations, using either discrete- or continuous-time models.[13,14]

In this section, different type of uncertainties, including price uncertainty, processing time uncertainty, and demand uncertainty, are studied. Given the advantage of Bertsimas and Sim's robust formulation, this approach is adopted in this paper, although comparisons with the other models are also provided in section 4.

3.1. Price Uncertainty. To apply Bertsimas and Sim's formulation for the price uncertainty in the objective function, the original objective is transformed to the following expression:

$$\begin{aligned} &\max\ \text{profit} \\ &\text{s.t. } \text{profit} - \sum_{s \in S_p, n} \widetilde{\text{price}}_s d_{s,n} + \\ &\qquad \sum_{s \in S_r} \widetilde{\text{price}}_s (\mathrm{STI}_s - \mathrm{STF}_s) \le 0 \end{aligned} \tag{25}$$

Thus, the uncertain price parameters $\widetilde{\text{price}}_s$ ($\widetilde{\text{price}}_s \in [\text{price}_s - \widehat{\text{price}}_s, \text{price}_s + \widehat{\text{price}}_s]$) appear in the LHS of the new constraint. The number of uncertain parameters in this constraint is equal to the number of uncertain prices. If the number of uncertain prices is k, then the budget parameter Γ^p takes a value in the range of $[0, k]$. To apply the robust formulation, this constraint is transformed to the following set of constraints (expressions 26–29):

$$\text{profit} - \sum_{s \in S_p, n} \widetilde{\text{price}}_s d_{s,n} + \sum_{s \in S_r} \widetilde{\text{price}}_s (\mathrm{STI}_s - \mathrm{STF}_s) + \Gamma^p z^p + \sum_s q_s^p \le 0 \tag{26}$$

$$z^p + q_s^p \ge \widehat{\text{price}}_s \sum_n d_{s,n}, \quad s \in S_p \tag{27}$$

$$z^p + q_s^p \ge \widehat{\text{price}}_s (\mathrm{STI}_s - \mathrm{STF}_s), \ s \in S_r \tag{28}$$

$$0 \le \Gamma^p \le k, \quad z^p \ge 0, \quad q_s^p \ge 0 \tag{29}$$

Here, only one constraint is subject to uncertainty, and the number of uncertain parameters is the number of uncertain prices. The constraints introduced correspond to the constraints given in problem P7. Because $d_{s,n} \ge 0$ and $\mathrm{STI}_s - \mathrm{STF}_s \ge 0$, no auxiliary variable is incorporated.

3.2. Processing Time Uncertainty. Let us consider that the processing times take values in the symmetric region of $\tilde{T}_{i,j} \in [T_{i,j} - \hat{T}_{i,j}, T_{i,j} + \hat{T}_{i,j}]$. The original duration constraint is reformulated as follows:

$$\mathrm{Ts}_{i,j,n} - \mathrm{Tf}_{i,j,n} + \alpha_{i,j}\mathrm{wv}_{i,j,n} + \beta_{i,j}b_{i,j,n} + \tilde{\theta}_{i,j}\mathrm{wv}_{i,j,n} \le 0 \tag{30}$$

In eq 30, $\alpha_{i,j}$ and $\beta_{i,j}$ are calculated based on the nominal processing time:

$$\alpha_{i,j} = \frac{2T_{i,j}}{3}$$

$$\beta_{i,j} = \frac{2T_{i,j}}{3(v_{i,j}^{\max} - v_{i,j}^{\min})}$$

$\mathrm{Tf}_{i,j,n}$ represents the lower bound on the finishing time of the task, instead of the exact finishing time as determined by the original duration constraint in Ierapetritou and Floudas.[12] In the reformulated constraint (eq 30), uncertainty is modeled through the last term, $\tilde{\theta}_{i,j}\mathrm{wv}_{i,j,n}$. Here, $\tilde{\theta}_{i,j}$ represents the variable part of the processing time parameter, so it takes a value in the range of $[-\hat{T}_{i,j}, \hat{T}_{i,j}]$ and its nominal value is 0. Because there is only one uncertain coefficient in the reformulated duration constraint, the budget parameter Γ^t for the corresponding robust formulation takes a value in the range of [0, 1].

$$\mathrm{Ts}_{i,j,n} - \mathrm{Tf}_{i,j,n} + \alpha_{i,j}\mathrm{wv}_{i,j,n} + \beta_{i,j}b_{i,j,n} + 0 \cdot \mathrm{wv}_{i,j,n} + \Gamma^t z_{i,j,n}^t + q_{i,j}^t \le 0 \tag{31}$$

$$z_{i,j,n}^t + q_{i,j}^t \ge \hat{T}_{i,j}\mathrm{wv}_{i,j,n} \tag{32}$$

$$0 \le \Gamma^t \le 1, \quad z_{i,j,n}^t \ge 0, \quad q_{i,j}^t \ge 0 \tag{33}$$

The constraints given in eqs 31–33 correspond to the constraints in problem P7, respectively. For every duration constraint, one additional variable $z_{i,j,n}^t$ and one constraint are added; for every uncertain parameter $\tilde{\theta}_{i,j}$, one additional variable is added. So, in total, the additional number of variables is the sum of the number of uncertain duration constraints and the number of uncertain processing time parameters.

3.3. Demand Uncertainty. The demand uncertainty appears on the right-hand side of the demand constraints,

$$\sum_n d_{s,n} \ge \tilde{r}_s \tag{34}$$

Assume that the demand parameter takes a value in the symmetric region $\tilde{r}_s \in [r_s - \hat{r}_s, r_s + \hat{r}_s]$. To apply the robust formulation, an additional binary variable with fixed value of 1 is added and the constraint is transformed as follows:

$$-\sum_{n} d_{s,n} + \tilde{r}_s \cdot 1 \leq 0 \tag{35}$$

Thus, the demand constraint has only one uncertain coefficient, and, thus, the budget parameter Γ^d takes a value in the range of [0, 1]. The following constraints (expressions 36–38) are then incorporated into the robust formulation:

$$-\sum_{n} d_{s,n} + r_s + \Gamma^d z^d + q_s^d \leq 0 \tag{36}$$

$$z^d + q_s^d \geq \hat{r}_s \tag{37}$$

$$0 \leq \Gamma^d \leq 1, \quad z^d \geq 0, \quad q_s^d \geq 0 \tag{38}$$

4. Examples

In all the examples presented in this section, the resulting MINLP problem is solved in GAMS using the DICOPT Solver and the MILP problems are solved in GAMS using the CPLEX 10.1 Solver in a Pentium personal computer (PC) (2.8 GHz, 1G RAM) running on the Windows XP operating system platform.

4.1. Example 1. This example is taken from Ierapetritou and Floudas[12] and involves the production of two products using three raw materials (Figure 1). Through this example, we are comparing the three different robust counterpart optimization formulations stated in the previous section, considering different types of uncertainties, and finally the problem is solved using the Bertsimas and Sim's robust formulation with a systematic consideration of all the uncertainties.

4.1.1. Price Uncertainty. In this case, bounded and symmetric uncertainty of raw material cost and product price is assumed. The scheduling horizon is 8 h, and eight event points are used in the continuous scheduling formulation presented in section 3. The nominal cost of all raw materials is 5, and the nominal prices of products 1 and 2 are 10 and 15, respectively. A 5% variability level is assumed for all the prices, and no infeasibility tolerance is considered in the robust formulation.

First, the deterministic schedule using all nominal price values is solved, and the optimal schedule has an optimal profit of 1088.75. Because the objective is profit maximization, the worst-case scenario corresponds to the minimum value of all product prices (9.5, 14.25) and the maximum value of the raw material costs (5.25, 5.25, 5.25). The resulting profit from Soyster's worst-case formulation is 959.56. The extended Ben-Tal's robust formulation for the MILP problem proposed by Lin et al.[10] is solved with a reliability level of 28.4% and 10%, which means that the probability that the constraint is violated is, at most, 28.4% and 10%. Finally, the robust formulation by Bertsimas and Sim is solved using different budget parameters between 0 and 5 (the number of uncertain prices). A detailed comparison of the different robust formulations is shown in Table 1 (the optimality gap in the CPLEX Solver is set as 0.1).

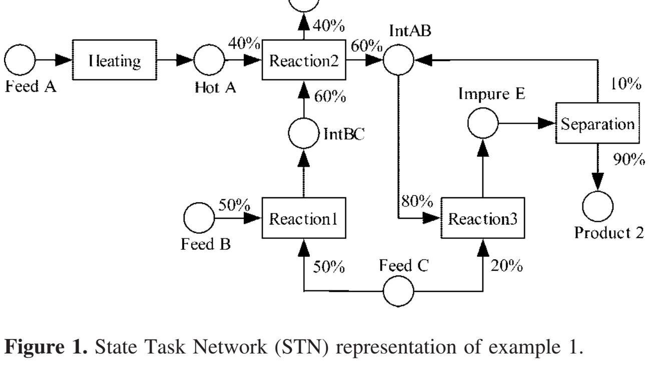

Figure 1. State Task Network (STN) representation of example 1.

As shown in Table 1, the result of the deterministic schedule using nominal parameter values is the same as the result of Bertsimas and Sim's formulation when the budget parameter is set as 0; the result of the Soyster's worst-case formulation is the same as that of Bertsimas and Sim's when the budget parameter takes its maximum value. Soyster's formulation has the same problem size as the nominal deterministic formulation, because the uncertain parameters are coefficients of positive variables and no auxiliary variable is added. Bertsimas and Sim's robust formulation is relatively more efficient, compared to Ben-Tal's formulation, because a MILP problem instead of a MINLP problem is solved. Moreover, when the maximum probability of constraint violation is the same, Bertsimas and Sim's robust formulation generates higher profit than Ben-Tal's formulation, which means that Ben-Tal's formulation is more conservative than Bertsimas and Sim's formulation.

4.1.2. Processing Time Uncertainty. In this case study, the scheduling horizon is 12 h and eight event points are used in the continuous scheduling formulation. Let us consider here that all the processing times are uncertain parameters and have a variability of 15%. In this case, we assume zero cost for all the raw materials, and product prices are set as 10 for both products 1 and 2.

Ben-Tal's formulation is studied with the reliability level set as 75% and 62.5%, and no infeasibility tolerance is assumed for all of the formulations. For Bertsimas and Sim's formulation, note that the maximum number of uncertain parameters is 1 for the duration constraints, so $\Gamma^t \in [0, 1]$.

The results in Table 2 lead to the same conclusions as in Table 1. In comparison to deterministic formulation, Soyster's formulation and Ben-Tal's formulation has the same variable and constraint number, because the uncertain parameters are coefficients of binary variables and no auxiliary variable is added. Bertsimas and Sim's formulation has more constraints and variables, but it is still more efficient than Ben-Tal's formulation, because the linearity of the formulation.

4.1.3. Demand Uncertainty. In this case, the scheduling horizon is set as 8 h and eight event points are used in the continuous scheduling formulation. Let us consider that both the demand of products 1 and 2 have a 50% variability level; they both take values in the range of [25, 75], and the nominal value is 50. A reliability level of 75% is set for the Ben-Tal's formulation, because a smaller reliability level (62.5%) causes the problem to be infeasible. The budget parameter value for Bertsimas and Sim's formulation takes a value in the range of [0, 1], because only one demand parameter is uncertain in the demand constraints.

The results in Table 3 illustrate similar trends as the previous case studies. Soyster's formulation becomes infeasible here, which means that the "worst-case" demand uncertainty cause infeasible schedules. Correspondingly, Bertsimas and Sim's formulation becomes infeasible when the budget parameter is set at the largest value. Therefore, the Bertsimas and Sim's robust formulation will be feasible for different budget parameters in the range of zero to the maximum uncertain coefficient of a constraint only if the worst-case feasibility is ensured. Furthermore, in this case, Soyster's formulation and Ben-Tal's formulation both have same problem size as the deterministic formulation, because the uncertain parameter is on the RHS of the constraint and it can be viewed as the coefficient of binary variable with a fixed value of 1.

Table 1. Comparison of the Robust Formulations for Price Uncertainty

					Bertsimas and Sim			
	nominal	Soyster	Ben-Tal		$\Gamma^p = 0$	$\Gamma^p = 2.5$	$\Gamma^p = 4.19$	$\Gamma^p = 5$
objective	1088.75	959.56	981.09	961.73	1088.75	989.63	967.44	959.56
upper probability of constraint violation			0.284	0.10	0.695	0.284	0.10	0.031
CPU time (s)	14.4	22.9	64.6	44.4	13.8	33.4	20.2	19.3
continuous variables	401	401	419		407			
binary variable	64	64	64		64			
constraints	884	884	899		889			

Table 2. Comparison of the Robust Formulations for Processing Time Uncertainty

					Bertsimas and Sim		
	nominal	Soyster	Ben-Tal		$\Gamma^t = 0$	$\Gamma^t = 0.5$	$\Gamma^t = 1$
objective	2657.9	2140.9	1676.1	1201.8	2657.9	2423.6	2140.9
upper probability of constraint violation			0.75	0.625	0.75	0.625	0.5
CPU time (s)	4.8	58.3	129.2	109.0	4.9	12.2	69.6
continuous variables	401	401	401		473		
binary variable	64	64	64		64		
constraints	884	884	884		948		

Table 3. Comparison of the Robust Formulations for Demand Uncertainty

					Bertsimas and Sim		
	nominal	Soyster	Ben-Tal		$\Gamma^d = 0$	$\Gamma^d = 0.5$	$\Gamma^d = 1$
objective	1088.75	infeasible	942.80	infeasible	1088.75	688.05	infeasible
upper probability of constraint violation			0.75	0.625	0.75	0.625	
CPU time (s)	29.0		60.9		28.7	181.8	
continuous variables	401	401	401		411		
binary variables	64	64	64		64		
constraints	884	884	884		893		

Table 4. Solution Data for Example 2 with All Uncertainties

Budget parameter (Γ^p,Γ^d, Γ^t)	(0,0,0)	(0.5,0.3,0.3)	(1,0.3,0.5)	(2,0.3,0.5)
objective	1088.75	615.5	431.3	402.6
CPU time (s)	26.7	237.2	231.3	330.5
continuous variables	494			
binary variables	64			
constraints	972			

Summarizing, three different types of uncertainties in scheduling have been studied and compared. From the results, we can see that the size of all the robust formulations do not increase much, because the increase in the number of constraints and variables is at the same scale as the number of the uncertain parameters. Moreover, because most decision variables in scheduling formulation are positive or binary, we can further reduce the number of auxiliary variables and boundary constraints for the auxiliary variables. Comparing the three different robust formulations, Soyster's worst-case formulation is the most conservative formulation and does not have the flexibility of adjusting the degree of conservatism. The other two robust counterpart formulations are able to adjust the solution robustness either through the constraint violation probability or the budget parameter. Bertsimas and Sim's formulation involve relative more constraints and continuous variables when addressing processing time and demand uncertainty, but it has more flexibility in controlling the degree of conservatism and also avoids the solution of mixed integer nonlinear optimization problem, thus the solution efficiency is greatly improved; although Ben-Tal's formulation can also adjust the degree of conservatism with the probability of constraint violation, it tends to be a more conservative formulation and thus more likely to generate infeasible problem; on the other hand, the feasibility of Bertsimas and Sim's formulation can be ensured when the feasibility of the worst-case formulation is satisfied. Therefore, Bertsimas and Sim's formulation will be adopted in our research as the method of generating robust preventive schedule, as shown in following subsection.

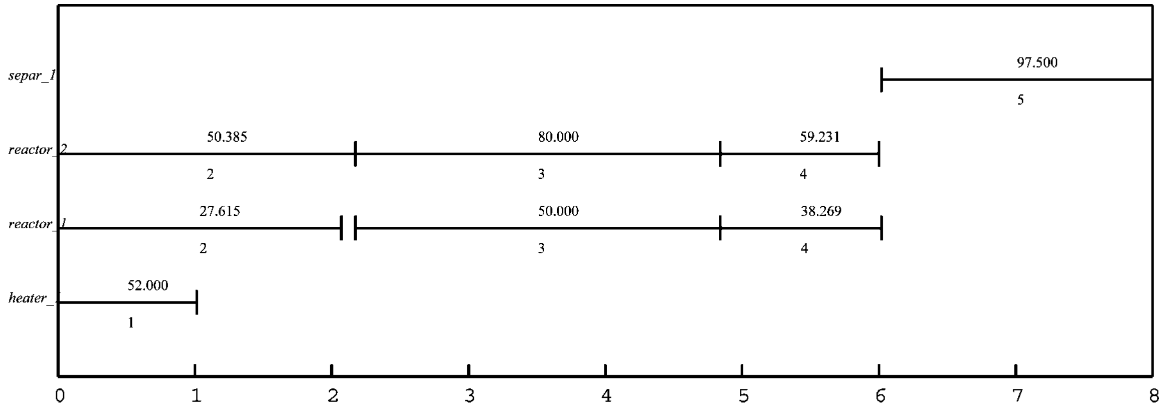

Figure 2. Nominal schedule for example 1 (*x*: hours, *y*: equipment).

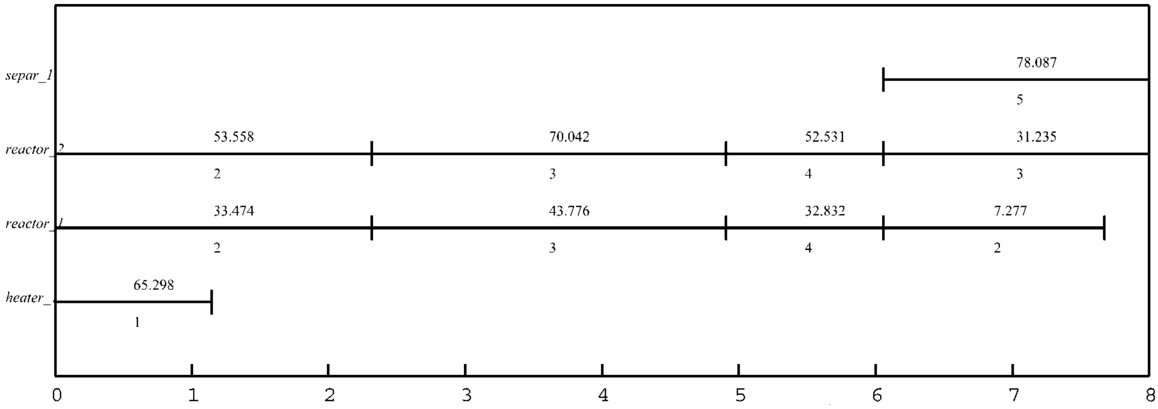

Figure 3. Robust schedule for example 1 ($\Gamma^p = 0.5$, $\Gamma^d = 0.3$, $\Gamma^t = 0.3$).

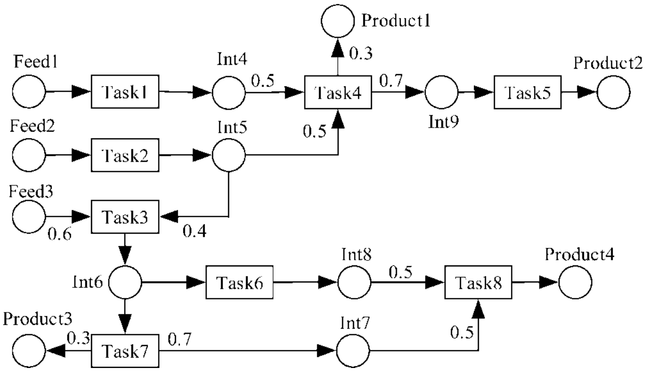

Figure 4. STN of example 2.

Table 5. Solution Data of Example 2

	Determinisitic formulation	Bertsimas and Sim's Robust formulation		
budget parameter (Γ^p,Γ^d, Γ^t)		(0,0,0)	(1,0.3,0.3)	(2,0.4,0.4)
objective	738835	738835	674567	634677
CPU time (s)	7.0	12.5	147.7	476.9
continuous variables	1471	1619		
binary variables	720	720		
constraints	2416	2554		

4.1.4. Systematically Considering All Uncertainties. Finally, we consider all uncertain parameters simultaneously including all processing times with variability 15%, the demand of products 1 and 2 with variability 50%, and the prices of products 1 and 2 with variability level 5%. The scheduling horizon is set as 8 h, and eight event points are used in the continuous scheduling formulation.

Several different budget parameter combinations are used to solve the problem considering all the uncertainties. The results are summarized in Table 4, which shows the relationship between the profit objective and the budget parameters: higher budget parameters result in a more conservative solution with larger feasibility but smaller profit. The profit of the nominal schedule with a budget parameter value of zero is 1088.75, and is shown in Figure 2. When we choose the budget parameter to be $\Gamma^p = 0.5$, $\Gamma^d = 0.3$ and $\Gamma^t = 0.3$, which means the corresponding price, demand, and duration constraints may be violated with a maximum probability of 61.3%, 67.5%, and 67.5%, respectively, the profit reduces to 615.5 (a 43.5% decrease). Because product 2 (P2) is more valuable than product 1 (P1), the production of the nominal schedule (Figure 2) leads to the production of 52 units of P1 and 87.5 units of P2, with the intent of producing more P2 for the largest profit. On the other hand, the robust schedule is shown in Figure 3, which has a tendency to generate a feasible schedule that covers a more uncertain range and has a production of 58 units of P1 and 70.3 units of P2, because the demand of both P1 and P2 is in the range of [25, 75]. Moreover, the objective of the robust schedule is to generate feasible operations, considering the

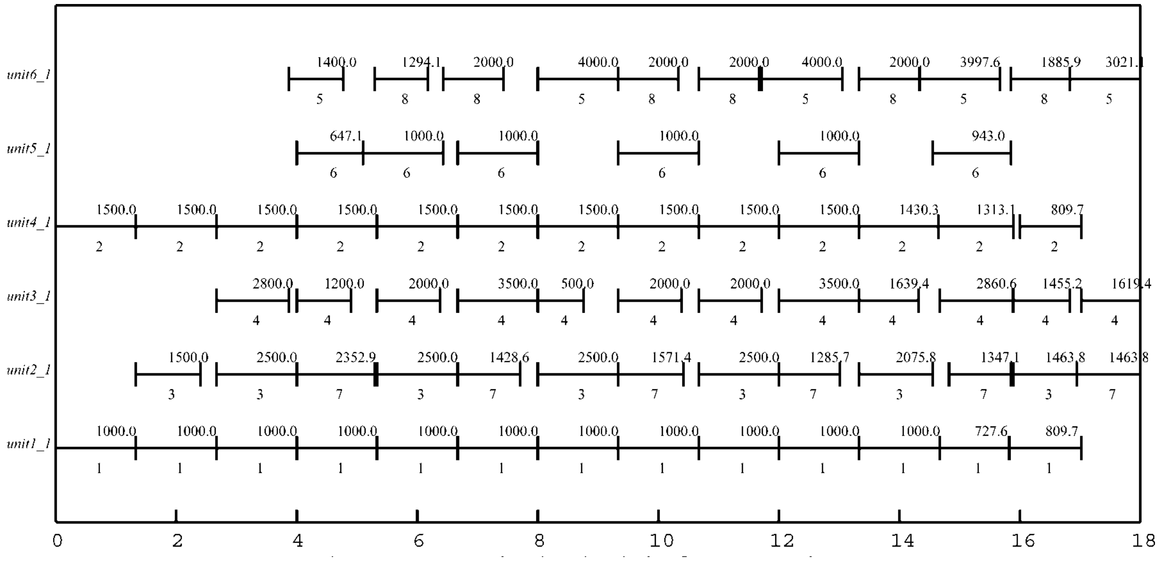

Figure 5. Nominal schedule for example 2

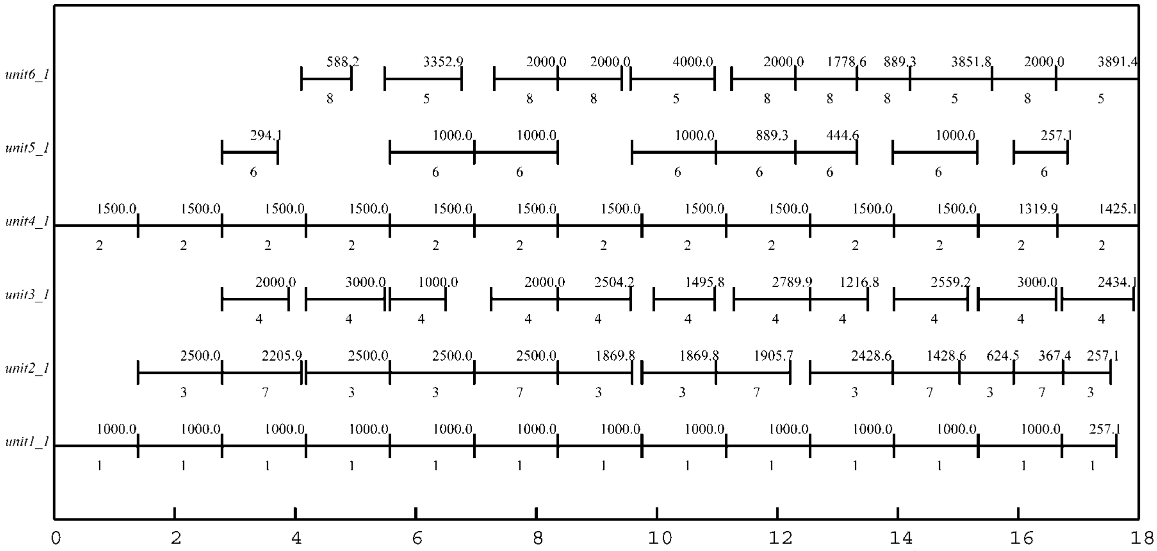

Figure 6. Robust schedule for example 2 ($\Gamma^p = 1$, $\Gamma^d = 0.3$, $\Gamma^t = 0.3$).

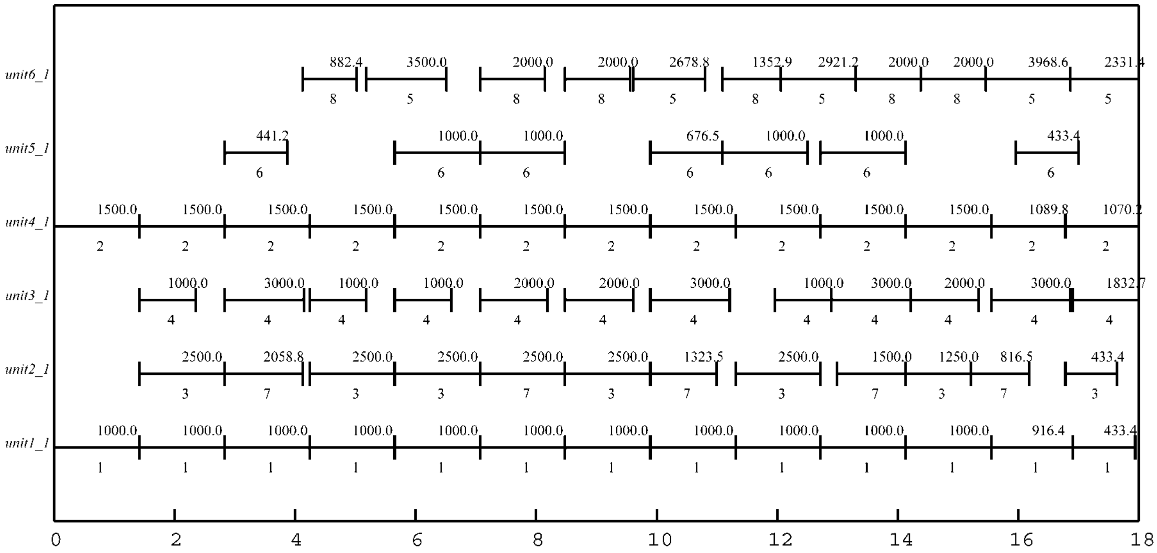

Figure 7. Robust schedule for example 2 ($\Gamma^p = 2$, $\Gamma^d = 0.4$, $\Gamma^t = 0.4$).

processing time variability (e.g., in the separation stage, a lesser amount of materials is processed in the robust schedule (78.087) than in the nominal schedule (97.5), such that the task will finish in the given time horizon, considering the processing time variability).

4.2. Example 2. This example is taken from Wu and Ierapetritou.[15] Here, four products are produced through eight tasks from three feeds. There are nine intermediates in the system as shown in Figure 4. In all, six different units are required for the entire process. This case study is considered because it is larger than the previous example to illustrate the scaleup of the robust-counterpart formulation.

The scheduling horizon is 18 h, and 15 event points are used in the continuous scheduling formulation. The nominal value of all the processing times is 1; the nominal price of products 1, 2, 3, and 4 (P1−P4) are 18, 19, 20, and 21, respectively; the nominal demand of products 1, 2, 3, and 4 are 6000, 8000, 2000, and 8000, respectively. For the uncertainty problem, we assume all the processing times have 20% variability level, all the product demands have 30% variability level, and all the product price have 10% variability level.

To generate robust schedules for this problem, the robust optimization formulation proposed by Bertsimas and Sim is also used here. Different budget parameter combinations are considered, as shown in Table 5 (the optimality gap in the CPLEX Solver is set as 0.01).

The deterministic scheduling formulation for example 2 is first solved, and the generated schedule is shown in Figure 5. The robust formulation with different budget parameter combinations then is used to generate the robust schedules. The robust schedule with a minimum budget parameter value of 0 is actually same as the nominal deterministic schedule. The other two robust schedules are shown in Figures 6 and 7. The production of products 1, 2, 3, and 4 in the nominal schedule are 7522, 16418, 2835, and 11180, respectively; the production in the robust schedule (Figure 6) are 7200, 15096, 2522, and 11256, respectively; and the production in the robust schedule (Figure 7) are 7150, 15400, 2459, and 1023, respectively, compared to the demand uncertainty: (6000, 8000, 2000, 8000)·(1 ± 30%). It can be seen that, as the budget parameter value increases, the production of the robust schedule has a tendency to reduce the production, to satisfy the duration requirement under the uncertain processing time with higher robustness; on the other hand, the total profit decreases. Note that when the budget parameters increase leading to more conservative solutions, the computation time becomes longer. This is due to the fact that the feasible space will become smaller as the robustness requirement increases.

5. Summary

To generate robust preventive schedules to address the different uncertainties in process scheduling problems, many efforts have been made in the past, especially in the direction of scenario-based stochastic scheduling. However, the scenario-based methodologies have a main drawback, in that they cannot avoid the exponential increase of the problem size when the number of parameters increases. Preventive scheduling that is based on robust counterpart optimization avoids this type of complexity. In this work, we studied three robust counterpart optimization formulations and compared their performance in uncertain scheduling. The results showed that the formulation proposed by Bertsimas and Sim is appropriate for uncertain scheduling problems, with its unique advantages that it does not increase the problem size substantially, it maintains its linearity, and it has the ability to control the degree of conservatism for every constraint and guarantee the feasibility for the robust optimization problem with the use of a budget parameter.

Nomenclature

M_l = index set for the uncertain coefficients in the lth constraint
$\tilde{a}_{lm}$ = true values of the coefficient parameter
a_{lm} = nominal values of the coefficient parameter
$\hat{a}_{lm}$ = variation amplitude of the coefficient parameter
u_m = auxiliary variable incorporated for robust formulation
ξ_{lm} = independent random variable symmetrically distributed in interval $[-1, 1]$
Ω_l = constant (with constraint violation probability$\kappa_l = e^{-\Omega_l^2/2}$
κ_l = constraint violation probability
z_{lm} = auxiliary variable incorporated for robust formulation
Γ_l = budget parameter
$\lfloor\Gamma_l\rfloor$ = biggest integer not greater than Γ_l
S_l = subset that contains $\lfloor\Gamma_l\rfloor$ uncertain parameters of the lth constraint
t_l = index that represents an additional uncertain parameter if Γ_l is not an integer
q_{lm} = dual variable
$i \in I$ = tasks
I_s = tasks which produce or consume state s
I_j = tasks which can be performed in unit j
$j \in J$ = units
J_i = units that are suitable for performing task i
$n \in N$ = event points representing the beginning of a task
$s \in S$ = states
S_p = states belonging to products
S_r = states belong to raw materials
price_s = price of state s
STI_s = initial amount of state s
STF_s = final amount of state s
$d_{s,n}$ = amount of state s delivered to the market at event point n
$\mathrm{wv}_{i,j,n}$ = binary, whether or not task i in unit j starts at event point n
$\mathrm{st}_{s,n}$ = continuous, amount of state s at event point n
$\rho^{\mathrm{P}}_{s,i}$ = proportion of state s produced by task i
$\rho^{\mathrm{C}}_{s,i}$ = proportion of state s consumed by task i
$b_{i,j,n}$ = amount of material undertaking task i in unit j at event point n
$\mathrm{st}_s^{\max}$ = available maximum storage capacity for state s
$v_{i,j}^{\min}$ = minimum amount of unit j when processing task i
$v_{i,j}^{\max}$ = maximum capacity of unit j when processing task i
r_s = market demand for state s at the end of the time horizon
$\mathrm{Tf}_{i,j,n}$ = time at which task i finishes in unit j while it starts at event point n
$\mathrm{Ts}_{i,j,n}$ = time at which task i starts in unit j at event point n
$\alpha_{i,j}$ = constant term of processing time of task i in unit j
$\beta_{i,j}$ = variable term of processing time of task i in unit j
H = time horizon

Acknowledgment

The authors gratefully acknowledge financial support from the National Science Foundation under Grants CTS 0625515 and 0224745.

Supporting Information Available: Process data for examples 1 and 2: task unit suitability, storage capacity, nominal processing times. This information is available free of charge via the Internet at http://pubs.acs.org.

Literature Cited

(1) Mulvey, J.; Vanderbei, R.; Zenios, S. Robust optimization of large scale systems. *Oper. Res.* **1995**, *43*, 264.

(2) Kouvelis, P.; Daniels, R. L.; Vairaktarakis, G. Robust scheduling of a two-machine flow shop with uncertain processing times. *IIE Trans.* **2000**, *32*, 421.

(3) Vin, J. P.; Ierapetritou, M. G. Robust short-term scheduling of multiproduct batch plants under demand uncertainty. *Ind. Eng. Chem. Res.* **2001**, *40*, 4543.

(4) Balasubramanian, J.; Grossmann, I. E. A novel branch and bound algorithm for scheduling flowshop plants with uncertain processing times. *Comput. Chem. Eng.* **2002**, *26*, 41.

(5) Jia, Z.; Ierapetritou, M. G. Uncertainty Analysis on the Right-Hand-Side for MILP problems. *AIChE J.* **2006**, *52*, 2486.

(6) Soyster, A. L. Convex programming with set-inclusive constraints and applications to inexact linear programming. *Oper. Res.* **1973**, *21*, 1154.

(7) Ben-Tal, A.; Nemirovski, A. Robust solutions to uncertain programs. *Oper. Res. Lett.* **1999**, *25*, 1.

(8) El-Ghaoui, L.; Oustry, F.; Lebret, H. Robust solutions to uncertain semidefinite programs. *SIAM J. Optim.* **1998**, *9*, 33.

(9) Bertsimas, D.; Sim, M. Robust Discrete optimization and Network Flows. *Math. Prog.* **2003**, *98*, 49.

(10) Lin, X.; Janak, S. L.; Floudas, C. A. A new robust optimization approach for scheduling under uncertainty: I. bounded uncertainty. *Comput. Chem. Eng.* **2004**, *28*, 1069.

(11) Ben-Tal, A.; Nemirovski, A. Robust solutions of Linear Programming problems contaminated with uncertain data. *Math. Prog.* **2000**, *88*, 411.

(12) Ierapetritou, M. G.; Floudas, C. A. Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. *Ind. Eng. Chem. Res.* **1998**, *37*, 4341.

(13) Floudas, C. A.; Lin, X. Continuous-time versus discrete-time approaches for scheduling of chemical processes: A review. *Comput. Chem. Eng.* **2004**, *28*, 2109.

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

(15) Wu, D.; Ierapetritou, M. G. Decomposition approaches for the efficient solution of short-term scheduling problems. *Comput. Chem. Eng.* **2003**, *27*, 1261.

Received for review October 22, 2007
Revised manuscript received March 6, 2008
Accepted March 7, 2008

IE071431U