ISSN 0040-5795, Theoretical Foundations of Chemical Engineering, 2017, Vol. 51, No. 6, pp. 893–909. © Pleiades Publishing, Ltd., 2017.

Mathematical Programming Techniques for Optimization under Uncertainty and Their Application in Process Systems Engineering[1]

I. E. Grossmann*, R. M. Apap, B. A. Calfa, P. Garcia-Herreros, and Q. Zhang

Carnegie Mellon University, Pittsburgh, PA 15213, USA

**e-mail: grossmann@cmu.edu*

Received May 10, 2017

Abstract—In this paper we give an overview of some of the advances that have taken place to address challenges in the area of optimization under uncertainty. We first describe the incorporation of recourse in robust optimization to reduce the conservative results obtained with this approach, and illustrate it with interruptible load in demand side management. Second, we describe computational strategies for effectively solving two stage programming problems, which is illustrated with supply chains under the risk of disruption. Third, we consider the use of historical data in stochastic programming to generate the probabilities and outcomes, and illustrate it with an application to process networks. Finally, we briefly describe multistage stochastic programming with both exogenous and endogenous uncertainties, which is applied to the design of oilfield infrastructures.

Keywords: robust optimization, stochastic programming, exogenous uncertainty, endogenous uncertainty, scenario generation

DOI: 10.1134/S0040579517060057

INTRODUCTION

Optimization under uncertainty has been an active area of research in process systems engineering [1–3]. A major decision in this area is whether one should rely on a robust optimization approach in which the emphasis is to guarantee feasibility over a specified uncertainty set, or whether one should use a stochastic programming approach in which first -stage decisions are made while anticipating that recourse actions can be implemented once the uncertainties are revealed. The robust optimization approach (or its generalization, chance constrained optimization) is usually more appropriate for short-term horizon problems in which feasibility is a major concern. On the other hand, stochastic programming is usually more appropriate for problems with long-term time horizons in which it is expected that recourse actions will be taken. Stochastic programming models, however, tend to be much more expensive to solve compared to robust optimization models. Furthermore, there is the question of how to specify the uncertainties (e.g., an intuitive guess or the use of historical data). Finally, it is essential to use as a basis an efficient deterministic model.

In this paper, we address the following major questions in optimization under uncertainty: (a) how to incorporate recourse in robust optimization, (b) how to reduce computational time when solving two-stage stochastic optimization problems, (c) how to make use of historical data in the generation of scenarios for stochastic programming, (d) how to handle exogenous (decision independent) and endogenous (decision dependent) uncertainties in multi-stage stochastic programming. As opposed to our recent paper in this area [3], we emphasize the modeling and application of specialized solution methods in industrial problems related to demand side management, supply chains, process networks and oilfields.

MODELING RECOURSE IN ROBUST OPTIMIZATION

Robust optimization [4] is one of the main approaches for incorporating uncertainty in optimization modeling. The uncertainty is specified in terms of an uncertainty set in which any point is a possible realization of the uncertainty. The major goal is to find a solution that is feasible for all possible realizations of the uncertainty while optimizing the objective function. Since the worst case of the uncertainty set is one of the possible realizations, a robust optimization model returns a solution that is optimal for this partic-

[1] The article is published in the original.

ular point. However, considering the worst case is often overly conservative. One can reduce the level of conservatism by adjusting the size of the uncertainty set as proposed by Bertsimas and Sim [5], in which a pre-specified "budget" parameter limits the number of uncertain parameters that can change at the same time. The general formulation for linear models is as follows:

$$\min_{x}\{c^T x : A(u)x \leq b \ \ \forall u \in U\}, \quad (1)$$

where the parameter u is uncertain and defined over the uncertainty set U in which we assume that the uncertainty budget is included. Eq. (1) corresponds to a semi-infinite programming problem, but can be simplified by considering the dual of each row in the matrix A (robust counterpart), which yields a finite dimensional optimization problem. A drawback with this approach is that specifying the uncertainty set U is not always trivial since the uncertainty set might not be well known, and especially because the user has to specify the uncertainty "budget" ranging from very conservative (all parameters vary independently) to less risk-averse (e.g., limiting the number of independent changes). Another reason for over-conservatism in traditional robust optimization is because recourse is not accounted for (i.e., implement decisions after the realization of the uncertainty), which may not be realistic in some cases, such as in problems involving investment and long-term contract decisions. Below we present a recent development in robust optimization that allows recourse to a certain extent, and demonstrate the effectiveness of this method by applying it to an industrial case study. For a recent work for accounting for recourse in a rigorous manner the reader is referred to Zhang et. al [6] who have established interesting theoretical connections between flexibility analysis and robust optimization.

The affine decision rule approach. Consider the following multi-stage (T stages) optimization problem under uncertainty,

$$\min_{x}\left\{c^T x_1 : A_1(u)x_1 + \sum_{t=2}^{T} A_t x_t(u) \leq b \ \ \forall u \in U\right\}, \quad (2)$$

where x_t is the vector of the t-th-stage variables. For simplicity, we assume that the objective function only depends on x_1 and that only matrix A_1 is uncertain. While x_1 does not depend on u, x_t for $t \geq 2$ are recourse variables and depend on the realization of the uncertainty. Note that x_t only depends on the u that are realized up to stage t.

The problem given in Eq. (2) cannot be solved as such since the set of possible functions for $x_t(u)$ is infinitely large. The idea in the Decision Rule approach [7], also referred to as Adjustable Robust Optimization [4], is to restrict oneself to a certain type of functions for $x_t(u)$, in particular to the set of affine functions. Hence, we set $x_t(u) = \alpha_t + B_t u$, and we obtain the following robust formulation by constraint-wise construction:

$$\min_{x_1,\alpha,B}\left\{c^T x_1 : \max_{u\in U}\left\{a_{1,i}^T(u)x_1 + \sum_{t=2}^{T} a_{ti}^T(\alpha_t + B_t u)\right\} \leq b_i \ \ \forall i\right\}, \quad (3)$$

which can be reformulated into a single-level problem for certain types of uncertainty sets by using techniques applying strong duality. This results in a robust counterpart formulation in which the decision variables are x_1, and the parameters for the affine decision rules are α and B. By applying the Decision Rule approach, the multi-stage problem is transformed into a single-stage problem to which the classic robust optimization reformulation is applied. Obviously, it is quite restrictive to only consider affine functions; however, this allows us to retain computational tractability and still account for recourse to a certain extent (for rigorous treatment of recourse see [2]).

Industrial case study. We consider the problem of providing interruptible load in an air separation plant [8]. We describe in this section the deterministic version of this problem, which is then extended to the adjustable robust optimization described in the previous section.

Specifically, we consider a power-intensive continuously operated plant that can produce a certain set of products, for which given demands have to be satisfied. There are inventory capacities for storable products, and additional products can be purchased at given costs. We assume that for fixed production, all production costs except for the cost of electricity are constant. In this way, for optimization purposes, the total operating cost only consists of the electricity cost and the cost of purchasing products. Electricity prices, which are time-sensitive, are assumed to be known for the scheduling horizon. Besides selling products, the plant can gain additional revenue from providing operating reserve in the form of interruptible load, which is capacity for load reduction that the grid operator can request from the plant in case of contingency. The load reduction is measured with respect to the plant's target power consumption. The interruptible load provider is rewarded regardless how much load reduction is actually required, which is uncertain.

The goal is to find a production schedule over a given time horizon that guarantees satisfaction of all product demand under every possible realization of the uncertainty, which lies in the actual demand for load reduction. The solution is considered optimal if it minimizes net operating cost for the worst case, where the net operating cost is primarily the electricity cost and product purchase cost minus the revenue from providing interruptible load. We distinguish between two types of decisions: here-and-now decisions which have to be made at the beginning and cannot be changed over the course of the scheduling horizon, and wait-and-see decisions which can be adjusted after realization of the uncertainty. Here-and-now decisions are the modes of operation, the target production rates for each product, and the committed purchase amounts for each product in each time period of the scheduling horizon. The wait-and-see decisions are changes in production rates and product purchases if load reduction is requested or has been requested in previous time periods.

To model this problem, we assume that the plant can operate in different operating modes, which represent operating states such as "off", "on" and "startup". The feasible region for each mode is defined by a union of convex subregions in the product space, and a linear electricity consumption function with respect to the production rates is given for each subregion. The key feature here is that every subregion has the form of a polytope. Such a mode is generally referred to as a Convex Region Surrogate (CRS) model. For complex processes, CRS models can be constructed by either using a model-based [9] or a data-driven approach [10].

At any point in time, the plant can only run in one operating mode. For a given mode, the operating point has to lie in either one of the convex subregions. Any point in a subregion can be represented as a convex combination of the vertices of the polytope. These relationships can be expressed by the following constraints:

$$PD_{it} = \sum_{m} \sum_{r \in R_m} \overline{PD}_{mrit} \quad \forall i, t \in \overline{T}, \tag{4a}$$

$$\overline{PD}_{mrit} = \sum_{j \in J_{mr}} \lambda_{mrjt} \nu_{mrji} \quad \forall m, r \in R_m,\ i, t \in \overline{T}, \tag{4b}$$

$$\overline{PD}_{mrit} = \sum_{j \in J_{mr}} \lambda_{mrjt} = \overline{y}_{mrt} \quad \forall m, r \in R_m,\ t \in \overline{T}, \tag{4c}$$

$$EC_t = \sum_{m} \sum_{r \in R_m} \left(\delta_{mr} \overline{y}_{mrt} + \sum_{i} \gamma_{mri} \overline{PD}_{mrit} \right) \quad \forall t \in \overline{T}, \tag{4d}$$

$$y_{mt} = \sum_{r \in R_m} \overline{y}_{mrt} \quad \forall m, t \in \overline{T}, \tag{4e}$$

$$\sum_{m} y_{mt} = 1 \quad \forall t \in \overline{T}, \tag{4f}$$

where R_m is the set of subregions in mode m, and J_{mr} is the set of vertices of subregion $r \in R_m$. The binary variable y_{mt} equals 1 if mode m is selected in time period t, whereas $\overline{y}_{mrt}$ equals 1 if subregion $r \in R_m$ is selected in time period t. The amount of product i produced in time period t is denoted by PD_{it}. Associated with PD_{it} is the disaggregated variable $\overline{PD}_{mrit}$ for subregion $r \in R_m$, which is expressed as a convex combination of the corresponding vertices, ν_{mrji}. The amount of electricity consumed, EC_t, is a linear function of PD_{it} with a constant δ_{mr} and coefficients γ_{mri} specific to the selected subregion.

A transition occurs when the system changes from one operating point to another. In particular, constraints have to be imposed on transitions between different operating modes, which is achieved through Eqs. (5)–(7). The binary variables $z_{mm't}$ equals 1 if and only if the plant switches from mode m to mode m' at time t, which is enforced by the following constraint:

$$\sum_{m' \in \overline{TR}_m} z_{m'm,t-1} - \sum_{m' \in TR_m} z_{mm',t-1} = y_{mt} - y_{m,t-1} \quad \forall m,\ t \in \overline{T}, \tag{5}$$

where $\overline{TR}_m = \{m' : (m', m) \in TR\}$ and $TR_m = \{m' : (m, m') \in TR\}$ with TR being the set of all possible mode-to-mode transitions.

The restriction that the plant has to remain in a certain mode for a minimum amount of time after transition is stated as follows:

$$y_{m't} \geq \sum_{k=1}^{\theta_{mm'}} z_{mm',t-k} \quad \forall (m, m') \in TR,\ t \in \overline{T}, \tag{6}$$

with $\theta_{mm'}$ being the minimum stay time in mode m' after switching to it from mode m.

For the predefined sequences, each defined as a fixed chain of transitions from mode m to mode m' to mode m'', we can specify a fixed stay time in mode m' by imposing the following constraint:

$$z_{mm',t-\overline{\theta}_{mm'm''}} = z_{m'm''t} \quad \forall (m, m', m'') \in SQ,\ t \in \overline{T}, \tag{7}$$

where SQ is the set of predefined sequences and $\overline{\theta}_{mm'm''}$ is the fixed stay time in mode m' in the corresponding sequence.

The plant produces a set of products, of which some may be storable. As stated in Eq. (8a), the inventory level for product i at time t, IV_{it}, is the inventory level at time $t-1$ plus the amount produced minus the amount sold, SL_{it}. Eq. (8b) sets bounds on the inventory levels, and Eq. (8c) states that also products purchased from other sources, denoted by PC_{it}, can be used to satisfy demand.

$$\mathrm{IV}_{it} = \mathrm{IV}_{i,t-1} + \mathrm{PD}_{it} - \mathrm{SL}_{it} \quad \forall i, t \in \bar{T}, \tag{8a}$$

$$\mathrm{IV}_{it}^{\min} \le \mathrm{IV}_{it} \le \mathrm{IV}_{it}^{\max} \quad \forall i, t \in \bar{T}, \tag{8b}$$

$$\mathrm{SL}_{it} + \mathrm{PC}_{it} = D_{it} \quad \forall i, t \in \bar{T}. \tag{8c}$$

Interruptible load can be seen as the capability of a plant to reduce its electricity load within a short amount of time. It can hence be used as an operating reserve resource to release the stress on the power grid in times of contingency. When interruptible load is provided, the plant still operates at its planned target production level, but has to be ready to respond to load reduction requests. When such a request actually occurs, the plant has to deviate from its target production rate such that the requested load reduction is achieved.

To model the provision of interruptible load, we first replace $\overline{\mathrm{PD}}_{mrit}$ by the following sum:

$$\overline{\mathrm{PD}}_{mrit} = \mathrm{PD}_{mrit} + \mathrm{PD}_{mrit} \quad \forall m, r \in R_m, i, t \in \bar{T}, \tag{9}$$

where PD_{mrit} is the target production rate and PD_{mrit} is the response decrease in production rate when load reduction is required, in which case PD_{mrit} takes a negative value. The reduction in power consumption associated with the decrease in production with respect to the target production rate has to be at least the amount of requested load reduction, LR_t, as stated in the following constraint:

$$\sum_{m} \sum_{r \in R_m} \sum_{i} \gamma_{mri} \mathrm{PD}_{mrit} \le -\mathrm{LR}_t \quad \forall t \in \bar{T}, \tag{10}$$

where LR_t is normally an uncertain parameter.

We further define a binary variable x_t, which equals 1 if interruptible load is provided in time period t. When interruptible load is provided, there may be lower and upper bounds on the provided amount as stated in the following:

$$\mathrm{IL}_t^{\min} x_t \le \mathrm{IL}_t \le \mathrm{IL}_t^{\max} x_t \quad \forall t \in \bar{T}, \tag{11}$$

where IL_t is the amount of interruptible load provided in time period t.

The scheduling problem is formulated for a given time horizon. For the problem to be well-defined, initial conditions are required, which are given in the following:

$$\mathrm{IV}_{i,0} = \mathrm{IV}_i^{\mathrm{ini}} \quad \forall i, \tag{12a}$$

$$y_{m,0} = y_m^{ini} \quad \forall m, \tag{12b}$$

$$z_{mm't} = z_{mm't}^{ini} \quad \forall (m, m') \in TR, \; -\theta^{\max} + 1 \le t \le -1, \tag{12c}$$

with $\theta^{\max} = \max\left(\max_{(m,m') \in TR} (\theta_{mm'}), \max_{(m,m'm'') \in SQ} (\bar{\theta}_{mm'm''}) \right)$, which defines for how far back in the past the mode switching information has to be provided.

The objective is to minimize the total net operating cost, TC, which is defined as the sum of the electricity cost and the product purchase cost minus the revenue gained from providing interruptible load, as stated in the following equation:

$$\mathrm{TC} = \sum_{t \in \bar{T}} \left(\alpha_t^{\mathrm{EC}} \mathrm{EC}_t + \sum_i \alpha_{it}^{\mathrm{PC}} \mathrm{PC}_{it} - \alpha_t^{\mathrm{IL}} \mathrm{IL}_t \right), \tag{13}$$

where α_t^{EC}, α_t^{PC}, and α_t^{IL} are price coefficients.

With LR_t being the uncertain parameter, an affinely adjustable robust counterpart of the model described above is formulated by applying the approach presented in the previous section. To adjust the extent of recourse, the parameter $\bar{\zeta}$ is introduced, denoting the number of uncertain parameters from previous time periods that are incorporated in the linear decision rules. If $\bar{\zeta} = 0$, only the uncertain parameter from the current time period appears in the decision rules; hence, the only possible recourse actions are the reduction in production rate when load reduction is requested and additional product purchase in the same time periods. If $\bar{\zeta} > 0$, i.e. uncertain parameters from previous time periods are also taken into account, lost production can also be made up by increasing production or purchase in time periods after the load reduction occurred. Furthermore, in order to reduce the level of conservatism, the budget uncertainty set proposed by Zhang et al. [11] is applied, which sets a limit to the number of time periods in which maximum reserve dispatch can occur.

Applying the proposed model to an industrial air separation case study, for which the data are provided by Praxair, the scheduling problem is solved for a one-week time horizon with an hourly time discretization. The uncertainty set is chosen such that request for maximum load reduction can happen up to 7 times a week, which is a fairly conservative assumption. In practice, the budget parameters can be chosen based on historical data; alternatively, depending on the market, there may be a strict limit on the number of times in which load reduction can be requested during a specific time horizon, which can be used to set the budgets.

The solution strongly depends on the extent of recourse that is considered in the model. With $\bar{\zeta} = 0$, a cost reduction of 1.2% is achieved compared to the case in which no interruptible load is provided. These cost savings further increase by more than 50% if $\bar{\zeta}$ is changed to 23. However, this improvement in the quality of the solution comes at the cost of deteriorating computational performance. In the case of $\bar{\zeta} = 0$,

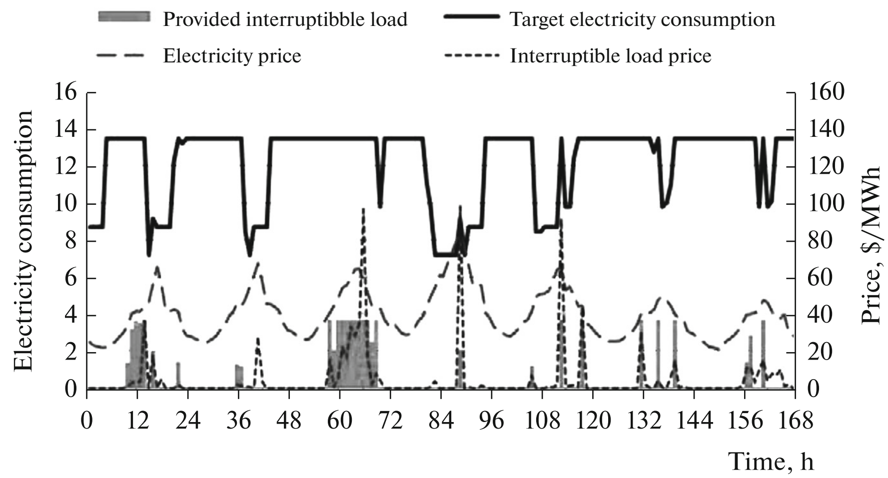

Fig. 1. Target electricity consumption profile and provided interruptible load for the case of $\overline{\zeta} = 23$, and price profiles.

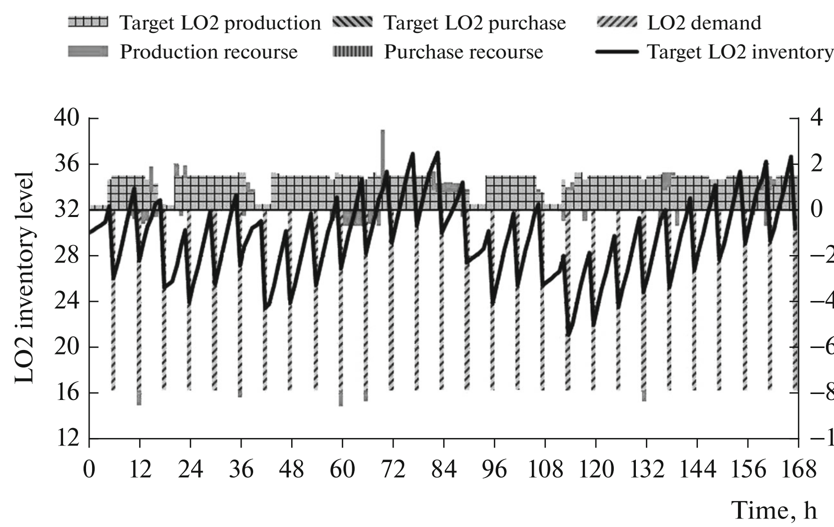

Fig. 2. Target and recourse LO2 flows and target inventory profile.

the model has 3282 binary variables, 82670 continuous variables, and 84604 constraints, while for $\overline{\zeta} =$ 23, the number of binary variables remains the same, but the numbers of continuous variables and constraints increase to 330242 and 325000, respectively. The computation times required to solve the models to 0.1% optimality gap using CPLEX 12.6 on an Intel Core i7-2600 machine at 3.40 GHz with 8 processors and 8 GB RAM are 185 s and 6 476 s. The parameter $\overline{\zeta}$ can be further increased (up to 167). However, computational experiments show only marginal improvement in the solution for $\overline{\zeta} > 23$; hence $\overline{\zeta} = 23$ is chosen as a good trade-off between level of conservatism and problem size.

The results for the case with greater extent of recourse, i.e. $\overline{\zeta} = 23$, are shown in Figs. 1 and 2. Along with the electricity and interruptible load prices, Fig. 1 shows the target load profile for the plant as well as the amount of interruptible load provided, which obviously has to be less than the target electricity consumption. For liquid oxygen (LO2), one of the products, Fig. 2 shows the inventory profile and the corresponding product flows as well as the cumulative recourse actions in terms of changes in production and purchase rates. Negative production recourse indicates time periods in which interruptible load is provided. One can see that the vast majority of the lost production is made up by increasing production after load reduction (positive production recourse).

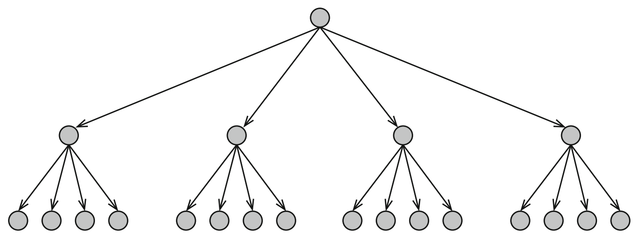

Fig. 3. Tree representation of scenarios in a stochastic program with three stages.

TWO-STAGE AND MULTISTAGE STOCHASTIC PROGRAMMING

Stochastic programming is the framework that models mathematical programs with uncertainty by optimizing the expected value over the possible realizations. In general, the expected value is computed by integrating over the set of uncertain parameters, which might be a challenging task. In the case of discrete uncertainty sets with finite support, the realizations can be characterized with a finite number of scenarios, simplifying the calculation of the expected value. Accordingly, stochastic programming is often regarded as a scenario-based approach for optimization under uncertainty [12]. The formulations can accommodate decision making at different stages according to the sequence in which uncertainty reveals. The stages imply a discrete time representation of the problem and establish the information of the uncertain parameters available at that time. The potential paths in which discrete uncertain parameters might evolve are represented in a scenario tree as shown in Fig. 3. In these trees, each node is a decision-making instance with known realization of the uncertain parameters up to the current state; potential future realizations are represented with branches from the given node.

The simplest stochastic programming formulation considers decisions that are made before uncertainty reveals. It is called single-stage stochastic programming or stochastic programming without recourse. Among the stochastic programs that consider recourse, the most widely used formulation is the Mixed-Integer Linear Program (MILP) with continuous recourse in a second stage. The two-stage stochastic programming formulation divides the decisions into two sets: here-and-now decisions that are made before uncertainty reveals, and wait-and -see decisions that are independent for each scenario. The typical formulation of a linear stochastic programming problem is presented in Eq. (14),

$$\min_{x_1 \in X_1} c^T x_1 + \mathrm{E}_{s\in S}[Q(x_1)],$$
$$\text{s. t. } Q(x_1) = \min_{x_2{}^s \in X_2^s} d_s^T x_2^s, \qquad (14)$$
$$\text{s. t. } W x_2^s \le h_s - T_s x_1,$$

where x_1 is the vector of first-stage decisions in mixed-integer polyhedral set X_1, x_2^s is the vector of second-stage (recourse) decisions in polyhedral set X_2^s, and s is the index for scenarios. An important property of the formulation presented in Eq. (14) is that the feasible region for the first-stage variables x_1 is a convex polyhedron [12]. Based on this property and assuming that the second-stage problems $Q(x, s)$ are bounded, an MILP reformulation that explicitly calculates the expected value can be derived by including all second-stage problems. This reformulation, the deterministic equivalent of the stochastic programming problem that is presented in Eq. (15), may lead to very large problem sizes if the number of scenarios is large.

$$\min c^T x_1 + \sum_{s\in S} p_s (d_s^T x_2^s),$$
$$\text{s. t. } W x_2^s \le h_s - T_s x_1 \quad \forall s \in S, \qquad (15)$$
$$x_1 \in X_1;\; x_2^s \in X_2^s.$$

The benefits of using a stochastic programming model can be quantified by the Value of the Stochastic Solution (VSS). The VSS is the difference between the expected value of the objective functions obtained from the stochastic formulation and a deterministic formulation that substitutes the uncertain parameters with their expectation. The expected value of the deterministic formulation is calculated by solving the problem, implementing the first-stage solution, and evaluating the scenarios with their optimal recourse. The model in Eq. (14) can also be extended to a multistage stochastic programming model. The tree corresponding to a three-stage problem has the same structure as the one shown in Fig. 3. The solution of multistage stochastic programming problems is considerably harder, and special care must be taken to avoid anticipating the uncertain parameters that have not been revealed. The general formulation of a three-stage stochastic programming problem is presented in Eq. (16),

$$\min_{x_1 \in X_1} c^T x_1 + \mathrm{E}_{s_2}[Q_2(x_1)],$$
$$\text{s. t. } Q_2(x_1) = \min_{x_2{}^s \in X_2^s} d_s^T x_2^s + \mathrm{E}_{s_3|s_2}[Q_3(x_2^s)],$$
$$\text{s. t. } W_s x_2^s \le h_s - T_s x_1, \qquad (16)$$
$$Q_3(x_2^s) = \min_{x_3{}^s \in X_3^s} f_s^T x_3^s,$$
$$\text{s. t. } V_s x_3^s \le g_s - U_s x_1 - W_s x_2^s,$$

One way of transforming the formulation in Eq. (16) into its deterministic equivalent is to generate a set of copied variables for each path from the root node to the branches, and introduce non-anticipativity constraints [13]. This reformulation has the advantage of being relatively easy to implement. The solution of large-scale stochastic programming problems is a challenging area of research due to the very large number of scenarios needed to model industrial problems and the rapid growth of scenarios in multistage stochastic programming. Methods frequently used to solve large stochastic programming problems leverage the scenario structure of the problems. The L-shaped method is the implementation of Benders decomposition to two-stage stochastic programming problems [14]. The method considers the first-stage variables as complicating and iterates between a relaxed master problem and subproblems that are solved by scenario. Lagrangean relaxation is also used in decomposition strategies for two-stage and multistage stochastic programming problems. The most common implementation generates sets of copied variables for each scenario and their corresponding non-anticipativity constraints. A Lagrangean relaxation of the problem is obtained by dualizing the non-anticipativity constraints, which makes the problem amenable to scenario decomposition. Other decomposition strategies are Progressive Hedging and Nested Decomposition procedures.

Example of two-stage stochastic programming. The supply chain design problem considered involves selecting distribution centers (DCs) among a set of candidate locations, determining their storage capacity for multiple commodities, and establishing the distribution strategy [15]. The objective is to minimize the sum of investment costs and expected distribution cost. Distribution costs are incurred during a finite time-horizon that is modeled as a sequence of time-periods. These costs include transportation from plant to DCs, storage of inventory at DCs, transportation from DCs to customers, and penalties for unsatisfied demands.

The DC candidate locations are assumed to have associated risks of disruption. The risk is characterized by a probability that represents the fraction of time that the potential DC is expected to be disrupted. For the set of potential DC locations, the possible combinations of active and disrupted locations give rise to a discrete set of scenarios regardless of the investment decisions. The scenarios determine the potential availability of DCs. Actual availability depends on the realization of scenarios and the investment decisions. The distribution strategy implies establishing demand assignments in all possible scenarios. Assignments are modeled with continuous variables to allow customers to be served from different DCs simultaneously. Customer demands are satisfied from active DCs according to the availability of inventory. Unsatisfied demands are subject to penalty costs. The expected cost of distribution is calculated from the distribution cost of each scenario according to its associated probability. DCs are assumed to follow a continuous review

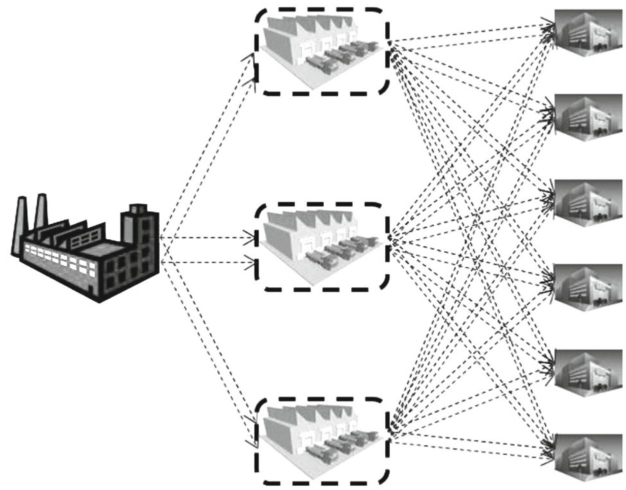

Fig. 4. General configuration of supply chain with supply plant, distribution centers and markets.

base-stock inventory policy with zero lead time. All cost coefficients are assumed to be known and deterministic.

The problem is formulated as a two-stage stochastic program. In the first stage, DCs are selected from a set of candidate locations and their capacities are established. In the second stage, the demand assignment decisions are made according to the selected DCs and random disruptions in each scenario.

The following notation is used in the formulation: the set of candidate locations for DCs is denoted by J, the set of customers is denoted by I, the set of scenarios is denoted by S, x_j is the binary variable deciding if a DC is open at candidate location j, $y_{s,j,i}$ represents the fraction of demand of customer i satisfied from location j in scenario s, $w_{s,i}$ represents the fraction of demand of customer i that is not satisfied in scenario s, v_j^w is the working inventory in location j, and v_j^{sf} is the safety inventory in location j. The parameters of the problem are: the number of time-periods in the design horizon (χ), the demand of customer i per time-period (D_i), the holding cost of a unit of inventory per time-period (h), the fixed-charge for opening DCs (f), the capacity cost for DCs per unit of inventory (c), the transportation cost per unit from plant to DC $j(a_j)$, the transportation cost per unit from DC j to customer $i(b_{j,i})$, the penalty cost per unit of unsatisfied demand (p), the probability of scenario $s(\pi_s)$, the maximum capacity of DCs ($V^{\max}$), the matrix indicating the availability of DC j in scenario $s(St_{s,j})$, and the array indicating the distinguishability of scenarios s and s' with respect to the investment in DC $j(NAA_{s,s',j})$.

When the transportation times are neglected from the formulation, the inventory can be brought to its base-stock level instantly after the placement of a replenishment order. In this sense, the inventory is found at its optimal level before costumer demands are revealed in every time-period. The objective function (17) minimizes the sum of investment costs of DCs, the expected cost of transportation from plants to DCs, the expected cost of storage in DCs, the expected cost of transportation from DCs to customers, and the expected cost of penalties.

$$\min \sum_{j\in J\setminus\{|J|\}} \left[F_j x_j + \sum_{k\in K} V_{j,k} c_{j,k} \right] + N \sum_{s\in S} \pi_s \sum_{j\in J} \left[\sum_{k\in K} \left(A_{j,k} \sum_{i\in I} D_{i,k} y_{s,j,i,k} + \sum_{i\in I} B_{j,i,k} D_{i,k} y_{s,j,i,k} \right) + \sum_{k\in K} H_k \left(c_{j,k} - \frac{1}{2} \sum_{i\in I} D_{i,k} y_{s,j,i,k} \right) \right]. \quad (17)$$

The optimization problem is subject to the following constraints:

$$\sum_{j\in \mathrm{DC}} y_{s,j,i,k} = 1 \quad \forall s\in S, i\in I, k\in K, \quad (18)$$

$$c_{j,k} - C^{\max} x_j \le 0 \quad \forall j\in J, k\in K, \quad (19)$$

$$\sum_{i\in I} D_i y_{s,j,i,k} - T_{s,j} c_{j,k} \le 0 \quad \forall s\in S, j\in J, k\in K, \quad (20)$$

$$x_j \in \{0,1\},\ 0 \le y_{s,j,i,k} \le T_{s,j},\ c_{j,k} \ge 0 \quad \forall s\in S, j\in J, i\in I, k\in K. \quad (21)$$

Constraints (18) ensure demand assignments for all scenarios. Constraints (19) bound the storage capacity of DCs. Constraints (20) ensure that customer assignments in every scenario are restricted by the inventory available at DCs; inventory availability at DCs depends on their capacity and the binary matrix ($T_{s,j}$) that indicates the realization of disruptions ($T_{s,j} = 0$) in the scenarios.

The example considered here includes: 1 production plant, 9 candidate locations for DCs, and 30 customers with demands for 2 commodities. The parameters of the instance were generated randomly. The candidate DCs have disruption probabilities between 2 and 10%. The number of scenarios in the full-space problem is $2^9 = 512$. The design is based on a time-horizon (N) of 365 days; on this time-scale, investment cost can be interpreted as annualized cost. The instance is used to illustrate the use of Benders decomposition, the benefits of strengthening the master problem, and the impact of solving a reduced subset of relevant scenarios. The selected relevant subset of scenarios includes scenarios with up to 4 simultaneous disruptions, for a total of 256 scenarios with probability equal to 99.99%.

The full-space model yields a total cost of \$ 7225447, while the reduced model yields a similar cost of \$ 722491. Both solutions predict the same design decisions and therefore the same investment cost. The largest difference in the results is in the expected cost of penalties. Both problems were solved to 0% optimality tolerance using the special Benders decomposition algorithm described in [15]. Both formulations involve 9 binary variables; the full-space model has 318479 constraints and 309263 continuous variables, while the reduced-space model has 159247 and 154639, respectively. The Multi-cut Benders solution times were 281 s and 151 s, respectively, while the Strengthened multi-cut Benders solution times were 176 s and 89 s, respectively. As can be seen in Fig. 5, the use of the strengthened multi-cut master problem reduces the solution time because it requires fewer iterations. The solution times for the full-space and the reduced instances without any decomposition strategy, using GUROBI 5.5.0, were 3349 s and 1684 s, respectively. The much smaller solution times with the decomposition algorithms demonstrate that the proposed methodology is effective to solve large-scale instances of high computational complexity.

DATA-DRIVEN APPROACHES TO SCENARIO GENERATION

Before solving a robust or stochastic optimization model in practice, one has to choose a proper model of the uncertainty. This is motivated by the fact that the quality of such model directly impacts the quality of the solution of the optimization problem. In this section, we discuss methods for uncertainty modeling that use historical and forecast data for scenario-based optimization frameworks (e.g., stochastic programming).

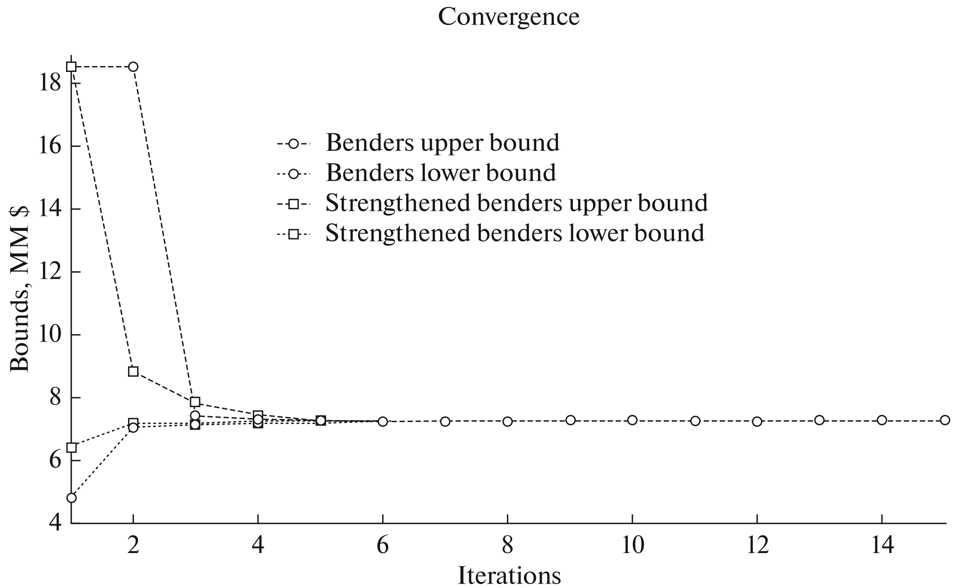

Fig. 5. Convergence of Benders algorithms for the full-space instance of the large-scale example.

Scenario trees are discrete representations of the probable outcomes of uncertain parameters, which are continuous random variables in most applications in process systems engineering. Also, a scenario tree is an input to a stochastic programming model. Therefore, systematically generating scenario trees that more accurately capture the true, but generally unknown structure of the uncertainty improves the quality of the solution to the original stochastic problem [16]. Given an initial structure of the tree (number of stages and number of branches from each node), data-driven scenario tree generation methods directly use available data to specify the probabilities of the branches and values of the outcomes.

In this section, we discuss the property-matching method of Høyland and Wallace [17] and an extension that addresses the potential under-determination of such a method. The property-matching method aims at generating scenario trees by matching statistical properties (e.g., moments and co-moments) calculated from the tree to the respective properties estimated from actual data (historical or forecast). Specifically, generating a scenario tree requires solving an optimization problem in which the objective function is an error measure (deviation of properties from the tree and data), and the decision variables are branch probabilities and node values. Consider the case of a two-stage scenario tree with N scenarios that is used to model a single uncertain parameter (see Fig. 6). The scenario probabilities and node values are represented by the vectors p and x, respectively.

The property-matching optimization model is given in Eq. (22). The set of statistical properties to be matched is denoted by S, and the target value for property s is given by Sval_s. Also, $f_s(\cdot,\cdot)$ is the mathematical expression of statistical property s calculated from the tree, and w_s is a weight vector.

$$\begin{aligned} &\min_{x,p} \sum_{s\in S} w_s (f_s(x,p) - \mathrm{Sval}_s)^2, \\ &\text{s.t.} \quad \sum_{s=1}^{N} p_s = 1, \\ &p_s \in [0,1] \quad \forall s = 1,\ldots,N. \end{aligned} \tag{22}$$

The optimization model in (22) may be over- or under-determined depending on the number of branches, targets, and uncertain parameters. For instance, as discussed in [18], for one uncertain parameter and four moments as targets, it can be calculated that we can only have a tree with $N = 2$ scenarios for a well-posed model. If we were to increase N to three or more, then this would result in an under-specified model. To overcome this under-specification issue and to allow more scenarios to be considered, a distribution-matching approach has been proposed and is briefly described as follows.

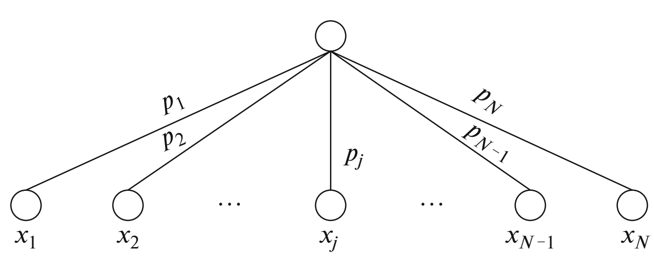

Fig. 6. Two-stage scenario tree for one uncertain parameter.

Table 1. Numerical results for the oilfield development planning problem

Problem Type	Total Expected NPV ($\$10^9$)		Optimality Gap	Solution Time (s)
	lower Bound	upper Bound		
Reduced Model	6.968	10.495	50.61%	40562
SSD	7.166	–	0.20%	41
LD	–	7.180		14

In addition to matching (co-)moments, points of the (empirical or estimated) marginal cumulative distribution function (CDF) are also matched, thus increasing the number of target values and balancing the difference between the number of variables and data points. Specifically, let s = CDF denote the marginal CDF property to be matched. Following the notation in Eq. (16), the functional form used in the distribution matching problem is given by Eq. (23).

$$f_{\mathrm{CDF}}(x,p) = \mathrm{CDF}(x_s) - \sum_{s'=1}^{s} p_{s'} \quad \forall s = 1,\ldots,N, \quad (23)$$

where $\mathrm{CDF}(\cdot)$ denotes the empirical or estimated marginal CDF of random variable x. Note that we must ensure that the node values are ordered due to the cumulative information being matched, e.g., $x_s \le x_{s+1}, \forall s = 1,\ldots,N-1$. The approach described above can be extended for the case of multiple uncertain parameters as well as autocorrelated uncertainties (stochastic processes), such as product demand and price. In the latter case, scenario tree generation is aided by time series forecasting [18].

Two-stage scenario tree generation example. Figure 7 shows a process network [18] consisting of a raw material A, an intermediate product B, finished products C and D (only product D can be stored), and plants P1, P2, and P3. Product C can also be purchased from a supplier, or in the case of multiple sites, it could be transferred from another site that also produces it. In this example, we solve a multiperiod production planning problem to determine the optimal flow rates in the network. The multiperiod problem is modeled as a two-stage stochastic program as follows. The first stage, or here-and-now variables, are all the variables in the model at the first time period, $t = 1$, whereas the second stage, or wait-and-see variables, are all the variables at the remaining time periods, $t > 1$.

The linear programming deterministic equivalent problem is as follows. The objective function in Eq. (24) to be maximized is the expected profit,

$$\max \sum_{s\in S} \pi_s \sum_{t\in T} \left[\begin{array}{l} \displaystyle\sum_{m\in \mathrm{FP}} \mathrm{SP}_{m,t} x^{\mathrm{sales}}_{m,t,s} - \sum_{f\in F} \mathrm{OPC}_{f,t} w^{\mathrm{rate}}_{f,t,s} - \sum_{\substack{m\in M:\\ \mathrm{MPUR}=1}} \mathrm{PC}_{m,t} w^{\mathrm{purch}}_{m,t,s} \\ \displaystyle - \sum_{\substack{m\in M:\\ \mathrm{MINV}=1}} \mathrm{IC}_{m,t} w^{\mathrm{inv}}_{m,t,s} - \sum_{m\in FP} \mathrm{PEN}_{m,t} \mathrm{slack}^{\mathrm{sales}}_{m,t,s} \\ \displaystyle - \sum_{f\in F} \mathrm{PEN}_{f,t} (\mathrm{slack}^{\mathrm{max,cap}}_{f,t,s} + \mathrm{slack}^{\mathrm{min,cap}}_{f,t,s}) \end{array} \right], \quad (24)$$

where π_s is the probability of scenario s, $\mathrm{SP}_{m,t}$ is the selling price of material m in period t, $\mathrm{OPC}_{f,t}$ is the operating cost of facility f in period t, $\mathrm{PC}_{m,t}$ is the purchase cost of material m in period t, $\mathrm{IC}_{m,t}$, is the inventory cost of material m in period t, and $\mathrm{PEN}_{m,t}$ and $\mathrm{PEN}_{f,t}$ denote the penalties associated with unmet demand and capacity violation. The decision variables include sales/purchase/inventory amounts $(x^{\mathrm{sales}}_{m,t,s}, x^{\mathrm{purch}}_{m,t,s}, w^{\mathrm{inv}}_{m,t,s})$, and inlet/outlet flow rates to/from facilities $(w^{\mathrm{rate}}_{f,t,s}, y^{\mathrm{rate}}_{f,t,s})$. The model constraints are given as follows,

$$w^{\mathrm{rate}}_{f,t,s} = \theta_{f,s} y^{\mathrm{rate}}_{f,t,s} \quad \forall f \in F, t \in T, s \in S, \quad (25a)$$

$$x^{\mathrm{sales}}_{\mathrm{C},t,s} = w^{\mathrm{rate}}_{\mathrm{P2},t,s} + x^{\mathrm{purch}}_{\mathrm{C},t,s} \quad \forall t \in T, s \in S, \quad (25b)$$

$$\begin{aligned} w^{\mathrm{inv}}_{\mathrm{D},t,s} &= w^{\mathrm{inv}}_{\mathrm{D},t-1,s} \\ &+ w^{\mathrm{rate}}_{\mathrm{P2},t,s} - x^{\mathrm{sales}}_{\mathrm{D},t,s} \quad \forall t \in T, s \in S, \end{aligned} \quad (25c)$$

$$w^{\mathrm{rate}}_{\mathrm{P1},t,s} = y^{\mathrm{rate}}_{\mathrm{P2},t,s} + y^{\mathrm{rate}}_{\mathrm{P3},t,s} \quad \forall t \in T, s \in S, \quad (25d)$$

$$x^{\mathrm{purch}}_{\mathrm{A},t,s} = y^{\mathrm{rate}}_{\mathrm{P1},t,s} \quad \forall t \in T, s \in S, \quad (25e)$$

$$\begin{aligned} x^{\mathrm{sales}}_{m,t,s} + \mathrm{slack}^{\mathrm{sales}}_{m,t,s} &= \xi_{m,t} \\ \forall m \in \mathrm{FP}, t &\in T, s \in S, \end{aligned} \quad (25f)$$

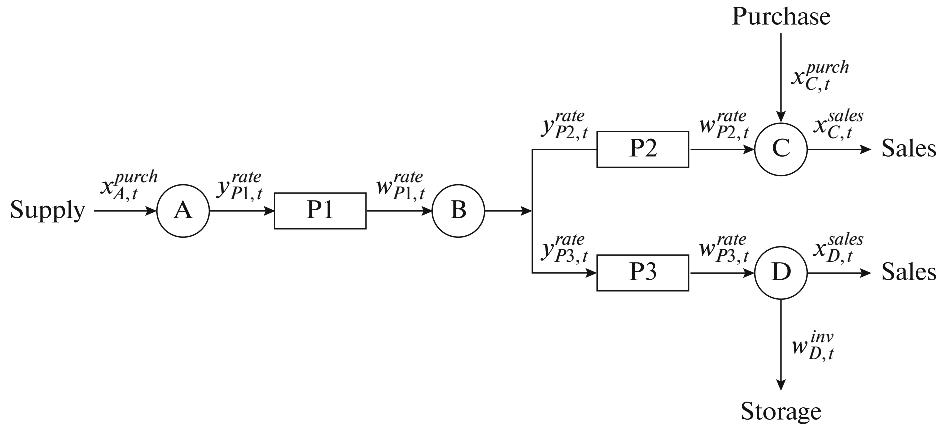

Fig. 7. Process network with uncertain yield for plant 1.

$$w_{f,t,s}^{\text{rate}} \le w_{f,t}^{\text{rate,max}} + \text{slack}_{f,t,s}^{\text{max,cap}} \quad \forall f \in F, t \in T, s \in S, \tag{25g}$$

$$w_{f,t,s}^{\text{rate}} \ge w_{f,t}^{\text{rate,min}} - \text{slack}_{f,t,s}^{\text{min,cap}} \quad \forall f \in F, t \in T, s \in S, \tag{25h}$$

$$w_{\text{D},t,s}^{\text{inv}} \le w_{\text{D},t}^{\text{inv,max}} \quad \forall t \in T, s \in S, \tag{25i}$$

$$x_{\text{A},t,s}^{\text{purch}} \le x_{\text{A},t,s}^{\text{purch,max}} \quad \forall t \in T, s \in S, \tag{25j}$$

where $\theta_{f,s}$, is the production yield, $\xi_{m,t}$ is the demand of finished products, $w_{f,t}^{\text{rate,max}}$ and $w_{f,t}^{\text{rate,min}}$ are maximum and minimum capacities of each facility, $w_{\text{D},t}^{\text{inv,max}}$ is the maximum inventory amount of product D, and $x_{\text{A},t,s}^{\text{purch,max}}$ is the maximum purchase amount of raw material A. Constraints (25a) relate the output flows with the input flows through the yield of each facility f, constraints (25b)–(25e) represent material and inventory balances, equations (25f) represent the demand satisfaction and slack variables are employed to account for possible unmet demand, constraints (25g)–(25j) are limitations in the flows, storage, raw material availability, and capacity violations, respectively. Non-anticipativity conditions (not shown) are also included in the model; they enforce that the decision variables take on the same value for all scenarios at $t = 1$ (e.g., $w_{f,1,s}^{\text{rate}} = w_{f,1,s'}^{\text{rate}}$ where the scenario $s \neq s'$).

The yield of plant P1 is the uncertain parameter and the distribution of its historical data is given in Fig. 8. The empirical CDF was obtained and approximated by a smooth function (generalized logistic function), i.e., $\widehat{\text{CDF}}(\cdot)$ in Eq. (17). In this example, $N = 5$ scenarios were considered, and the targets include the first four moments and CDF information. The scenario generation problem is a nonlinear program with 10 variables and 14 constraints. It was modeled in AIMMS 3.13 and solved with IPOPT 3.10.1 using AIMMS' multi-start module in less than one second. Fig. 9 shows optimized scenario probabilities of the generated scenario tree. Note that the profile captures the shape of the distribution of the original data, including tail effects.

MULTISTAGE STOCHASTIC PROGRAMMING UNDER ENDOGENOUS AND EXOGENOUS UNCERTAINTIES

Uncertain parameters can be classified into two categories depending upon the way the uncertainty is resolved: exogenous, where realizations occur independently of the decisions; and endogenous, where the realizations are affected by the decisions. In the endogenous case, the decisions may affect the timing of the realizations or their underlying probability distribution [19]. In this section, we consider the first class of endogenous uncertainty where decisions affect the timing of realizations. In the context of process systems engineering, exogenous uncertainties often correspond to market uncertainties, such as oil prices. Endogenous uncertainties are technical uncertainties, such as oilfield size, that are only resolved after a decision is made at a given point in time (e.g., to drill a particular oilfield). Surprisingly, although many problems contain both types of uncertainties, optimization under both types has been largely unexplored in the literature.

In the case where the uncertainty is purely exogenous, the shape of the scenario tree is known in advance, since exogenous realizations occur automatically in each time period. This is shown in Fig. 10 with the standard form of the tree, as well as its alternative representation which gives each scenario a unique set of nodes.

In the case of endogenous uncertainty, however, the shape of the scenario tree is conditional, since the timing of realizations depends on the process decisions. As shown in Fig. 11, we use a superstructure

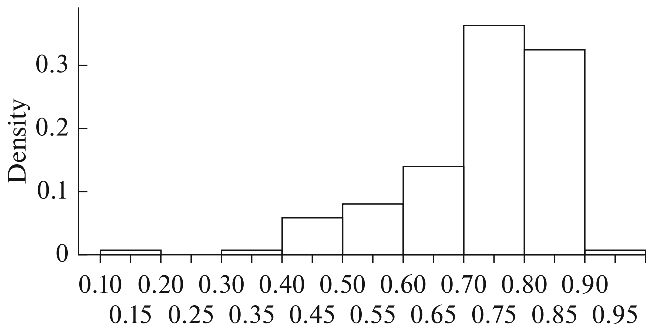

Fig. 8. Distribution of the historical data for the production yield of plant P1.

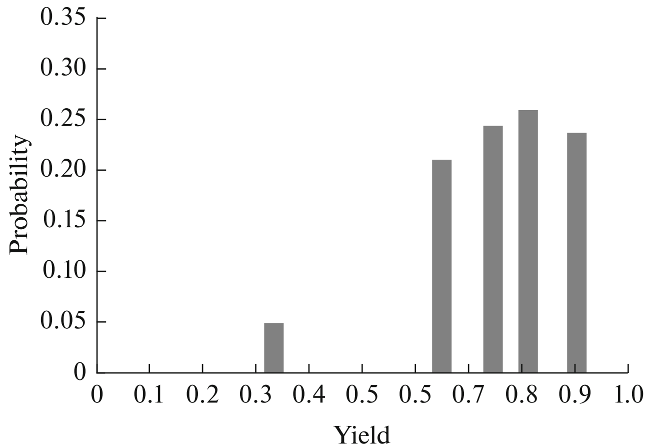

Fig. 9. Profile of scenario probabilities obtained with the distribution-matching approach.

form of the alternative tree in order to capture all possible outcomes [20].

For simplicity, we assume that the scenarios correspond to all possible combinations of realizations of the uncertain parameters. In other words, the set of scenarios in the exogenous scenario tree corresponds to a Cartesian product over the sets of realizations for the exogenous parameters (denoted by R_X), and, similarly, the set of scenarios in the endogenous scenario tree corresponds to a Cartesian product over the sets of realizations for the endogenous parameters (denoted by R_N). For the case of both endogenous and exogenous parameters, we generate the set of scenarios by simply taking the Cartesian product of all possible combinations of realizations of the endogenous parameters and all possible combinations of realizations of the exogenous parameters, $R_N \times R_X$. This is equivalent to copying the exogenous scenario tree for each possible combination of realizations of the endogenous parameters. We then link these exogenous trees by adding first-period and endogenous non-anticipativity constraints, thereby producing what we refer to as a 'composite' scenario tree as shown in Fig. 12.

The general model for this class of problems, originally proposed by Goel and Grossmann [21], is a mixed-integer linear disjunctive program. A simplified, compact form of the updated model [20] is given in Eq. (26)–(34).

$$\min_{y} \phi = \sum_{s\in S} p^s \sum_{t\in T} c_t^s y_t^s, \tag{26}$$

$$\text{s.t.} \sum_{\tau=1}^{t} A_{\tau,t}^s y_\tau^s \le a_t^s \quad \forall t \in T, s \in S, \tag{27}$$

$$y_1^s = y_1^{s'} \quad \forall (s,s') \in \mathrm{SP}_F, \tag{28}$$

$$y_{t+1}^s = y_{t+1}^{s'} \quad \forall (t,s,s') \in \mathrm{SP}_X, \tag{29}$$

$$y_{t+1}^s = y_{t+1}^{s'} \quad \forall (t,s,s') \in \mathrm{SP}_N, t \in T_E, \tag{30}$$

$$\begin{bmatrix} Z_t^{s,s'} \\ y_{t+1}^s = y_{t+1}^{s'},\ t < |T| \end{bmatrix} \vee \left[\neg Z_t^{s,s'}\right] \quad \forall (t,s,s') \in SP_N, t \in T_C, \tag{31}$$

$$Z_t^{s,s'} \Leftrightarrow F(y_1^s, y_2^s, \ldots, y_t^s) \quad \forall (t,s,s') \in \mathrm{SP}_N, t \in T_C, \tag{32}$$

$$y_t^s \in Y_t^s \quad \forall t \in T, s \in S, \tag{33}$$

$$Z_t^{s,s'} \in \{\text{True}, \text{False}\} \quad \forall (t,s,s') \in \mathrm{SP}_N, t \in T_C. \tag{34}$$

The objective function, Eq. (26), minimizes the total expected cost associated with decisions x_t^s, weighted by the probability of each scenario, p^s. Eq. (27) represents constraints that govern decisions x_t^s and link decisions across time periods. First-period non-anticipativity constraints (NACs) are given by Eq. (28), exogenous NACs are given by Eq. (29), and disjunctive constraint (30) conditionally enforces endogenous NACs based on the value of Boolean variable $Z_t^{s,s'}$. The value of $Z_t^{s,s'}$ is determined by an uncertainty resolution rule in Eq. (31).

Because the number of NACs grows exponentially as the number of time periods, uncertain parameters, or realizations increases, most problems of practical interest are too large to be solved directly with commercial MILP solvers. In order to eliminate redundant NACs, a number of theoretical properties have been proposed by Goel and Grossmann [21], Gupta and Grossmann [22] and Apap and Grossmann [20] based on the concepts of symmetry, adjacency, transitivity, and scenario grouping. These properties drastically reduce the number of constraints; however, the resulting model is often still intractable, and special solution approaches are required.

One effective approach is Lagrangean decomposition (LD), in which the complicating non-anticipativ-

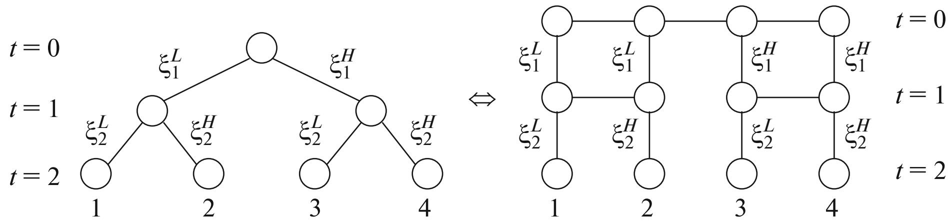

Fig. 10. An exogenous scenario tree and its alternative representation.

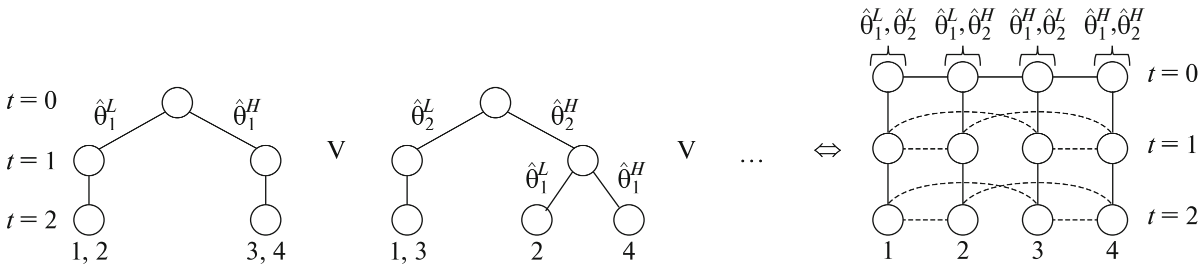

Fig. 11. A superstructure representation for endogenous scenario trees.

ity constraints are dualized in order to decompose the problem into independent scenario subproblems that can be solved in parallel. More recently, a sequential scenario decomposition heuristic (SSD) has been proposed that involves sequentially solving endogenous MILP subproblems to determine the integer decisions, fixing these decisions to satisfy the first-period and exogenous NACs, and then solving the original problem as an LP [20].

Oilfield planning example. We consider a modified form of the MILP described in [23] for maximizing the total expected NPV in the development planning of an offshore oilfield. There are 3 oilfields, 3 potential Floating Production Storage and Offloading vessels (FPSOs), and 9 possible field-FPSO connections. A total of 30 wells can be drilled over a 5-year planning horizon: 7 for field I, 11 for field II, and 12 for field III. There is also a 3-year lead time for FPSO construction and a 1-year lead time for FPSO expansion. Fields II and III have a known recoverable oil volume (size); however, the size of field I is uncertain. Specifically, there are 2 possible realizations for the size of field I, both with equal probabilities. Note that this is an *endogenous* uncertainty, since the size cannot be realized until we drill the field and begin producing from it. The oil and gas prices are also uncertain, with 2 possible realizations with equal probabilities in each time period. These are *exogenous* uncertainties, since the oil and gas prices are market values that will be realized automatically in each time period. These prices are assumed to be correlated. The network superstructure for this problem instance is shown in Fig. 13a.

We have 2 possible combinations of realizations for the endogenous parameters and 32 possible combinations of realizations for the exogenous parameters. By considering all possible combinations of realizations of the uncertain parameters, we generate 64 scenarios with equal probabilities. This gives rise to a 6-stage, mixed-integer linear stochastic programming problem with 333249 constraints, 70465 continuous variables, and 7360 binary variables. After applying the theoretical reduction properties [20], there are 124980 con-

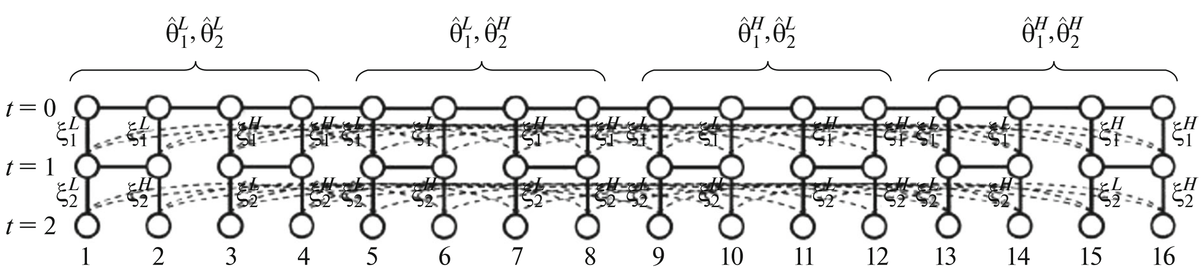

Fig. 12. A 'composite' scenario tree for endogenous and exogenous realizations.

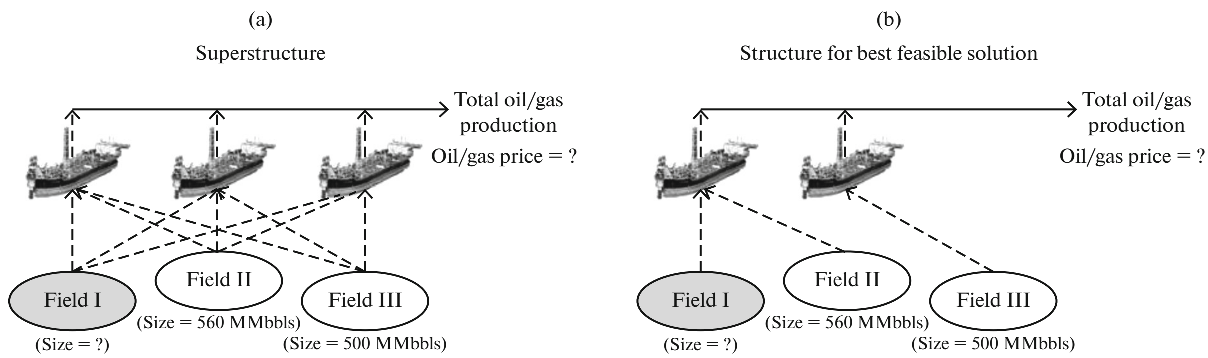

Fig. 13. Network structure for the oilfield development planning problem.

straints (a 62% reduction), 70465 continuous variables, and 7000 binary variables. The problem was modeled in GAMS 24.3.3 and solved with CPLEX 12.6.0.1 on a machine with a 2.93 GHz Intel Core i7 CPU and 12 GB of RAM.

Table summarizes the results for the example problem. In the case of solving the reduced model directly, the optimality gap cannot be improved past 50% after more than 11 hours. In contrast, SSD finds a high-quality feasible solution ($7.166 billion) in only 41 seconds. The LD algorithm uses this lower bound and finds a high-quality upper bound ($7.180 billion) in 14 seconds. This implies that the SSD solution is within 0.20% of the optimum.

The network structure corresponding to the best feasible solution ($7.166 billion, as obtained by the SSD heuristic) is shown in Fig. 13 (b). This solution indicates that we begin installing all necessary infrastructure in the first year. This includes FPSO I and FPSO II, as well as the following field-FPSO connections: field I to FPSO I, field II to FPSO I, and field III to FPSO II. Notice that due to the inherent risk in the size of field I, FPSO I is shared among fields I and II rather than devoting a separate FPSO solely to field I.

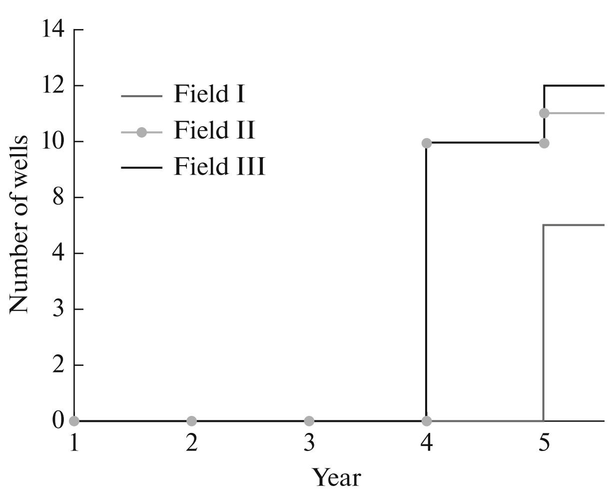

Fig. 14. Drilling schedule for the best feasible solution of the oilfield development planning problem.

The corresponding drilling schedule is shown in Fig. 14. Since it takes 3 years for the FPSOs to be fully operational, drilling cannot begin until the fourth year. For Field II, we drill 10 wells in year 4 and 1 well in year 5. Similarly for Field III, we drill 10 wells in year 4 and 2 wells in year 5. For Field I, we instead wait until year 5 and then drill 7 wells. The strategy here is to drill fields of known sizes first (as this carries less risk), and then drill the field with an uncertain size.

CONCLUSIONS

In this paper, we have given an overview of several major challenges in optimization under uncertainty. We have shown that recourse can be accounted for in robust optimization with linear Decision Rules, which has the effect of producing less conservative solutions. This has been illustrated on a demand side management problem for an air separation plant with interruptible load. To effectively solve two-stage stochastic optimization problems, we showed that it is very important to reduce the number of scenarios, tighten the MILP formulation, and apply a decomposition scheme like Benders decomposition with multiple cuts. We have illustrated the application of these strategies in the design of supply chains under disruptions. We have also shown that to avoid assigning arbitrary probabilities and outcomes in scenario trees, historical data can be incorporated to generate these scenarios by using moment matching supplemented by matching cumulative distribution function values. We have illustrated this capability in the design of process networks with historical information of uncertain yields. Finally, we have shown that both exogenous and endogenous uncertainties can be effectively handled in multi-stage stochastic programming by relying on theoretical properties to reduce the number of non-anticipativity constraints, and applying Lagrangean decomposition and fast heuristics. In this case, the example of the design and planning of an oilfield was considered.

ACKNOWLEDGMENTS

The authors would like to acknowledge financial support from NSF Grant No. 1159443, Praxair, Dow Chemical and the Center for Advanced Process Decision-making.

NOTATION

MODELING RECOURSE IN ROBUST OPTIMIZATION

u	uncertain parameter
U	uncertainty set
T	number of stages
x_t	vector of the -th-stage variables
α	parameter for the affine decision rules
B	parameter for the affine decision rules
R_m	set of subregions in mode m
J_{mr}	set of vertices of subregion $r \in R_m$
PD_{it}	amount of product i produced in time period t
EC_t	amount of electricity consumed
TR	set of all possible mode-to-mode transitions
$\theta_{mm'}$	minimum stay time in mode m' after switching to it from mode m.
SQ	set of predefined sequences
$\bar{\theta}_{mm'm''}$	fixed stay time in mode m'
$\mathrm{IV}_{i,t-1}$	inventory level at time $t-1$
SL_{it}	amount of product i sold in time period t
PC_{it}	products purchased from other sources
PD_{mrit}	target production rate
PD_{mrit}	response decrease in production rate when load reduction is required
LR_t	amount of requested load reduction
IL_t	amount of interruptible load provided in time period t
TC	total net operating cost
α_t^{EC}, α_t^{PC}, α_t^{IL}	price coefficients

TWO-STAGE AND MULTISTAGE STOCHASTIC PROGRAMMING

X_1	mixed-integer polyhedral set
x_1	vector of first-stage decisions in set X_1
X_2^s	polyhedral set
x_2^s	vector of second-stage (recourse) decisions in set X_2^s
J	set of candidate locations for distribution centers
I	set of customers
S	set of scenarios
x_j	binary variable deciding if a distribution center is open at candidate location j
$y_{s,j,i}$	fraction of demand of customer i satisfied from location j in scenario s
$w_{s,i}$	fraction of demand of customer i that is not satisfied in scenario s
v_j^w	working inventory in location j
v_j^{sf}	safety inventory in location j
χ	number of time-periods in the design horizon
D_i	demand of customer i per time-period
h	holding cost of a unit of inventory per time-period
f	fixed-charge for opening distribution centers
c	capacity cost for distribution centers per unit of inventory
a_j	transportation cost per unit from plant to distribution center j
$b_{j,i}$	transportation cost per unit from distribution center j to customer i
p	penalty cost per unit of unsatisfied demand
π_s	probability of scenario s
$V^{\max}$	maximum capacity of distribution centers
$St_{s,j}$	matrix indicating the availability of distribution center j in scenario s
$\mathrm{NAA}_{s,s',j}$	array indicating the distinguishability of scenarios s and s' with respect to the investment n distribution center j.

DATA-DRIVEN APPROACHES TO SCENARIO GENERATION

p	scenario probabilities vector
x	node values vector
S	set of statistical properties
Sval_s	target value for property s
$f_s(\cdot,\cdot)$	the mathematical expression of statistical property s calculated from the tree
w_s	weight vector
$\mathrm{CDF}(\cdot)$	empirical or estimated marginal cumulative distribution function of random variable x.

Two-stage scenario tree generation example

π_s	probability of scenario s
$SP_{m,t}$	selling price of material m in period t
$OPC_{f,t}$	operating cost of facility f in period t
$PC_{m,t}$	purchase cost of material m in period t
$IC_{m,t}$	inventory cost of material m in period t
$PEN_{m,t}$, $PEN_{f,t}$	penalties associated with unmet demand and capacity violationpenalties associated with unmet demand and capacity violation
$x^{sales}_{m,t,s}$, $x^{purch}_{m,t,s}$, $w^{inv}_{m,t,s}$	sales/purchase/inventory amounts
$w^{rate}_{f,t,s}$, $y^{rate}_{f,t,s}$	inlet/outlet flow rates to/from facilities
$\theta_{f,s}$	the production yield
$\xi_{m,t}$	demand of finished products
$w^{rate,max}_{f,t}$, $w^{rate,min}_{f,t}$	maximum and minimum capacities of each facility
$w^{inv,max}_{D,t}$	maximum inventory amount of product D
$x^{purch,max}_{A,t,s}$	maximum purchase amount of raw material A

MULTISTAGE STOCHASTIC PROGRAMMING UNDER ENDOGENOUS AND EXOGENOUS UNCERTAINTIES

R_X	exogenous parameters
R_N	endogenous parameters
x_t^s	decisions
p^s	probability of scenario.

SUBSCRIPTS AND SUPERSCRIPTS MODELING RECOURSE IN ROBUST OPTIMIZATION

m, m', m''	operating mode
t	time period
i	product.

TWO-STAGE AND MULTISTAGE STOCHASTIC PROGRAMMING

s, s'	index for scenarios
j	candidate location
i	customer

DATA-DRIVEN APPROACHES TO SCENARIO GENERATION

s	property

Two-stage scenario tree generation example

s, s'	index for scenarios
m	material
t	time period
f	facility
D	product
A	raw material

MULTISTAGE STOCHASTIC PROGRAMMING UNDER ENDOGENOUS AND EXOGENOUS UNCERTAINTIES

s, s'	index for scenarios
t	time period

REFERENCES

1. Sahinidis, N.V., Optimization under uncertainty: State-of-the-art and opportunities, *Comput. Chem. Eng.*, 2004, vol. 28, nos. 6–7, pp. 971–983. doi 10.1016/j.compchemeng.2003.09.017
2. Grossmann, I.E., Calfa, B.A., and Garcia-Herreros, P., Evolution of concepts and models for quantifying resiliency and flexibility of chemical processes, *Comput. Chem. Eng.*, 2014, vol. 70, pp. 22–34. doi 10.1016/j.compchemeng.2013.12.013
3. Grossmann, I.E., Apap, R.M., Calfa, B.A., Garcia-Herreros, P., and Zhang, Q., Recent advances in mathematical programming techniques for the optimization of process systems under uncertainty, *Comput. Chem. Eng.*, 2016, vol. 91, pp. 3–14. doi 10.1016/j.compchemeng.2016.03.002
4. Ben-Tal, A., El Ghaoui, L., and Nemirovski, A., *Robust Optimization*, Princeton: Princeton Univ. Press, 2009.
5. Bertsimas, D. and Sim, M., The price of robustness, *Oper. Res.*, 2004, vol. 52, no. 1, pp. 35–53. doi 10.1287/opre.1030.0065
6. Zhang, Q., Grossmann, I.E., and Lima, R.M., On the relation between flexibility analysis and robust optimization for linear systems, *AIChE J.*, 2016, vol. 62, no. 9, pp. 3109–3123. doi 10.1002/aic.15221
7. Kuhn, D., Wiesemann, W., and Georghiou, A., Primal and dual linear decision rules in stochastic and robust optimization, *Math. Program.*, 2011, vol. 130, no. 1, pp. 177–209. doi 10.1007/s10107-009-0331-4
8. Zhang, Q., Morari, M.F., Grossmann, I.E., Sundaramoorthy, A., and Pinto, J.M., An adjustable robust optimization approach to scheduling of continuous industrial processes providing interruptible load, *Comput. Chem. Eng.*, 2016, vol. 86, pp. 106–119. doi 10.1016/j.compchemeng.2015.12.018
9. Sung, C. and Maravelias, C.T., A projection-based method for production planning of multiproduct facil-

ities, *AIChE J.*, 2009, vol. 55, no. 10, pp. 2614–2630. doi 10.1002/aic.11845

10. Zhang, Q., Grossmann, I.E., Sundaramoorthy, A., and Pinto, J.M., Data-driven construction of convex region surrogate models, *Optim. Eng.*, 2016, vol. 17, no. 2, pp. 289–332. doi 10.1007/s11081-015-9288-8

11. Zhang, Q., Heuberger, C.F., Grossmann, I.E., Sundaramoorthy, A., and Pinto, J.M., Air separation with cryogenic energy storage: Optimal scheduling considering electric energy and reserve markets, *AIChE J.*, 2015, vol. 61, no. 5, pp. 1547–1558. doi 10.1002/aic.14730

12. Birge, J.A. and Louveaux, F., *Introduction to Stochastic Programming*, Springer Series in Operations Research and Financial Engineering, New York: Springer-Verlag, 2011, 2nd ed.

13. Ruszczyński, A., Decomposition methods in stochastic programming, *Math. Program.*, 1997, vol. 79, nos. 1–3, pp. 333–353. doi 10.1007/BF02614323

14. Van Slyke, R.M. and Wets, R., *L*-shaped linear programs with applications to optimal control and stochastic programming, *SIAM J. Appl. Math.*, 1969, vol. 17, no. 4, pp. 638–663. doi 10.1137/0117061

15. Garcia-Herreros, P., Wassick, J.M., and Grossmann, I.E., Design of resilient supply chains with risk of facility disruptions, *Ind. Eng. Chem. Res.*, 2014, vol. 53, pp. 17240–17251. doi 10.1021/ie5004174

16. King, A.J. and Wallace, S.W., *Modeling with Stochastic Programming*, Springer Series in Operations Research and Financial Engineering, New York: Springer-Verlag, 2012.

17. Høyland, K. and Wallace, S.W., Generating scenario trees for multistage decision problems, *Manage. Sci.*, 2001, vol. 47, no. 2, pp. 295–307. doi 10.1287/mnsc.47.2.295.9834

18. Calfa, B.A., Agarwal, A., Grossmann, I.E., and Wassick, J.M., Data-driven multi-stage scenario tree generation via statistical property and distribution matching, *Comput. Chem. Eng.*, 2014, vol. 68, no. 1, pp. 7–23. doi 10.1016/j.compchemeng.2014.04.012

19. Jonsbråten, T.W., Optimization models for petroleum field exploitation, *Doctor of Economics Dissertation*, Stavanger, Norway: Norwegian School of Economics and Business Administration, 1998.

20. Apap, R.M. and Grossmann, I.E., Models and computational strategies for multistage stochastic programming under endogenous and exogenous uncertainties, *Comput. Chem. Eng.*, 2017, vol. 103, pp. 233–274. doi doi 10.1016/j.compchemeng.2016.11.011

21. Goel, V. and Grossmann, I.E., A class of stochastic programs with decision dependent uncertainty, *Math. Program.*, 2006, vol. 108, nos. 2–3, pp. 355–394. doi 10.1007/s10107-006-0715-7

22. Gupta, V. and Grossmann, I.E., Solution strategies for multistage stochastic programming with endogenous uncertainties, *Comput. Chem. Eng.*, 2011, vol. 35, pp. 2235–2247. doi 10.1016/j.compchemeng.2010.11.013

23. Gupta, V. and Grossmann, I.E., A new decomposition algorithm for multistage stochastic programs with endogenous uncertainties, *Comput. Chem. Eng.*, 2014, vol. 62, pp. 62–79. doi 10.1016/j.compchemeng.2013.11.011