Optimization and Engineering manuscript No.
(will be inserted by the editor)

Enhancing explainability of stochastic programming solutions via scenario and recourse reduction

Tushar Rathi · Rishabh Gupta · Jose M. Pinto · Qi Zhang

Received: date / Accepted: date

Abstract Stochastic programming (SP) is a well-studied framework for modeling optimization problems under uncertainty. However, despite the significant advancements in solving large SP models, they are not widely used in industrial practice, often because SP solutions are difficult to understand and hence not trusted by the user. Unlike deterministic optimization models, SP models generally involve recourse variables that can take different values for different scenarios (i.e. uncertainty realizations), which makes interpreting their solutions a challenge when large numbers of scenarios and recourse variables are considered. In this work, we propose scenario and recourse reduction methods that can help enhance the explainability of SP solutions. Focusing on two-stage linear SP, the goal is to build reduced models, with much smaller sets of scenarios and recourse variables, that are easier to analyze yet still capture the key features of the original problems. Specifically, we explicitly search for reduced models that generate the same or close to the same first-stage decisions as the original SP models. The efficacy of the proposed methods is demonstrated in computational case studies involving problems of industrial relevance and size.

Keywords Stochastic programming · Explainability · Scenario reduction · Recourse reduction

1 Introduction

Stochastic programming (SP) (Birge and Louveaux, 2011) is a powerful tool for modeling optimization problems under uncertainty. The last few decades have seen tremendous advancement in terms of improving the tractability of large-scale SP models as well as identifying new applications that can be modeled using

Tushar Rathi, Rishabh Gupta, and Qi Zhang
Department of Chemical Engineering and Materials Science, University of Minnesota, Minneapolis, MN 55455, USA
E-mail: qizh@umn.edu

Jose M. Pinto
Linde Digital Americas, Linde plc, Danbury, CT 06810, USA

the SP paradigm. However, despite the progress made, SP is scarcely used in industrial practice. One of the main reasons for this is the inherent difficulty in understanding SP solutions. In SP, the uncertainty is commonly represented by a set of discrete scenarios, and it may require a large number of scenarios to approximate the underlying probability distribution with a high level of accuracy. In addition, SP models may involve recourse decisions that vary based on the chosen scenarios, which makes it challenging for the user to see the relationships between the different decisions and their impact on the objective function. With an ever-increasing emphasis on accounting for uncertainty in decision making, we believe that a structured explainability paradigm is needed to complement the existing SP methodology that can enhance its applicability to real-world problems.

The idea of improving the interpretability of SP solutions is inspired by a similar line of research in the field of machine learning (ML), popularly termed explainable artificial intelligence (XAI). Modern ML models, such as deep neural networks, can exhibit immense predictive power but tend to be complex and opaque; hence, they are also often referred to as black-box models. However, in many applications, the interpretability of the suggested predictions is as important as the predictions themselves; thus, the need to develop interpretable models or post hoc methodologies that can explain the outputs of ML models has recently gained significant attention. Post hoc XAI methods can be model-specific or model-agnostic. One example of a model-specific approach is the TreeSHAP method (Lundberg and Lee, 2017) that is particularly designed for tree-based models like random forests (Breiman, 2001). Model-agnostic explainability methods include local interpretable model-agnostic explanations (LIME) (Ribeiro et al., 2016) and counterfactual explanations (Wachter et al., 2017). Many XAI approaches utilize surrogate or simplified, more interpretable models to explain the output predictions. For a general overview of existing XAI techniques, we refer the reader to some of the many recent perspectives and reviews on this topic (Rudin, 2019; Arrieta et al., 2020; Belle and Papantonis, 2021).

Compared to most ML models, optimization models generally have a higher degree of interpretability as they consist of constraints that are typically derived explicitly from physical laws or logical expressions. However, although the model formulation may be easily interpretable, the process (or algorithm) of obtaining the optimal solution often is not. In fact, for all practical purposes, an optimization algorithm can be viewed as a black box. Especially in large-scale optimization involving many interacting variables and parameters, the complexity can be overwhelming and therefore limit the user's ability to understand why the provided solution is optimal. Existing sensitivity analysis methods can help better understand the solution by quantifying the impact of model parameters on the objective function (Ward and Wendell, 1990); however, this inherently local analysis often fails to provide satisfactory explanations in complex problems. Recently, Bertsimas and Stellato (2021) proposed a method for interpreting the optimal solution to an optimization problem via classification trees trained on different model instances. But for the most part, the interpretability of optimization models is very much an open research topic. With regard to SP, researchers have developed metrics such as the value of stochastic solution (Birge and Louveaux, 2011) and the expected value of perfect information to quantify the quality of an SP solution relative to a deterministic one. However, they do not offer any physical intuition or reasoning behind the choice of solution. To the best of our knowledge, so far, no systematic approach has been developed that specifically aims to improve the explainability of SP solutions.

In this work, we present an initial attempt in developing structured methods that aid the process of understanding SP solutions. We consider a general two-stage linear stochastic program of the following form:

$$\begin{aligned}
\underset{x, y_1, \dots, y_{|\mathcal{S}|}}{\text{minimize}} \quad & c^\top x + \sum_{s \in \mathcal{S}} p_s q_s^\top y_s \\
\text{subject to} \quad & Ax \leq b \\
& T_s x + W_s y_s \leq h_s \quad \forall\, s \in \mathcal{S} \\
& x \geq 0 \\
& y_s \geq 0 \quad \forall\, s \in \mathcal{S},
\end{aligned} \tag{SP}$$

where x and y denote the first- and second-stage variables, respectively. While the first-stage cost vector c and constraint parameters A and b are assumed to be known, the second-stage costs q and constraint parameters T, W, and h are generally considered to be uncertain. The uncertainty is represented by a set of discrete scenarios $\mathcal{S}$, with each scenario s given by the corresponding possible realization of the uncertainty (q_s, T_s, W_s, h_s) and the probability p_s. While the first-stage decisions are made before the realization of the uncertainty, the second-stage decisions, also called recourse decisions, can be made after the true values of the uncertain parameters are revealed. To model this sequential decision-making process, the formulation incorporates a separate y_s for each scenario s such that the recourse decisions can vary across the scenarios. The objective is to minimize the total expected cost.

The complexity of a two-stage SP and hence the difficulty of interpreting its solution increases with the numbers of scenarios and recourse variables, even when the corresponding deterministic problem is relatively easy to comprehend. However, we often observe that large SP problems can be approximated using a much smaller set of scenarios. Also, in many instances, we may only need a small subset of recourse variables to achieve the minimum expected cost. While we do not know those subsets of scenarios and recourse variables a priori, we have the opportunity to identify them after obtaining the optimal solution, which could allow us to create a significantly reduced, much more interpretable model. Following this idea, we propose two explainability methods that aim to reduce the complexity and hence enhance the explainability of an SP solution: (i) *Scenario clustering and reduction* based on similarity in the optimal recourse decisions, which reduces the number of scenarios and helps identify the features of the uncertainty that are the most relevant for the optimal SP solution. (ii) *Recourse reduction*, which identifies a set of principal recourse variables required to achieve the same optimal first-stage decisions while reaching the same or close to the same optimal value as the original SP model.

The literature that is most closely related to our work is the one on scenario reduction for SP (Dupačová et al., 2003). Existing scenario reduction techniques apply moment matching (Høyland and Wallace, 2001; Zhang and He, 2022; Bounitsis et al., 2022), transportation metrics such as Wasserstein and Sinkhorn distances (Li and Floudas, 2014; Kammammettu and Li, 2023), clustering (Latorre et al., 2007; Beraldi and Bruni, 2014; Feng and Ryan, 2013; Keutchayan et al., 2021;

Medina-Gonzalez et al., 2020), or optimization-based methods (Hewitt et al., 2022; Bertsimas and Mundru, 2022). Often, a combination of these concepts is used to achieve a reduced scenario set of desired cardinality and properties that closely resemble the distribution governing the original set of scenarios. Here, the main goal is to reduce the computational complexity of the SP problem; as such, it serves a different purpose than our proposed scenario reduction method aimed at enhancing explainability. For the same reason, the traditional scenario reduction is performed prior to solving the SP problem, while our method is a post hoc procedure. Similarly, limiting the choice of recourse variables is another common strategy for reducing the computational complexity of an SP problem. For example, multistage SP models are often approximated using two-stage formulations (Balasubramanian and Grossmann, 2004; Patriksson et al., 2015), where all recourse variables beyond the second stage are considered second-stage variables. In the extreme case, one can also consider all recourse variables to be first-stage (i.e. here-and-now) decisions, which leads to a static SP problem. Same as the traditional scenario reduction, this kind of recourse reduction is performed a priori and typically done in a heuristic fashion. In contrast, our proposed recourse reduction method is applied after obtaining the SP solution and follows a more rigorous approach.

The remainder of this paper is organized as follows. In Section 2, we present the proposed scenario clustering and reduction methods, followed by the development of the proposed recourse reduction approach in Section 3. To demonstrate the efficacy of the proposed methods, we conduct three computational case studies with applications relevant for the industrial gas industry. The results from these case studies are presented in Section 4. Finally, we close with some concluding remarks in Section 5.

2 Scenario clustering and reduction

The basic premise for the proposed methods is that we have already solved (SP) and obtained an optimal solution (x^*, y^*) with the corresponding optimal objective function value z^*. Naturally, one would now investigate this SP solution to check its validity and better understand why this particular solution is optimal. To aid this process, we develop scenario and recourse reduction methods to obtain a less complex SP model that can be more easily analyzed as well as to gain insights into the relationships between the uncertainty and the optimal decisions.

To reduce the number of scenarios, we follow the two-step procedure shown in Fig. 1. Given the full set of scenarios, we first apply k-means clustering to obtain clusters where the scenarios from the same cluster exhibit similar recourse decisions. Then, given these scenario clusters, we solve a scenario reduction problem to obtain a representative scenario for each cluster. This leads to a reduced SP model that only considers the chosen representative scenarios, and we construct it such that it has the same or close to the same optimal first-stage solution as the original SP model. The details of the scenario clustering and reduction methods are provided in the following subsections.

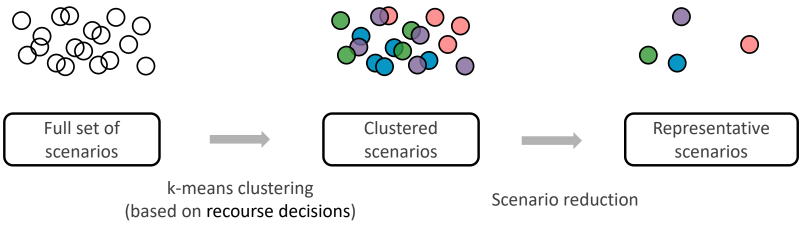


Fig. 1: Using k-means clustering, the full set of scenarios is first partitioned into k clusters based on the recourse decisions. Then, one representative scenario is selected from each cluster by solving the proposed scenario reduction problem.

2.1 Recourse-based scenario clustering

The proposed scenario clustering serves two main purposes: obtaining scenario clusters to be used in the subsequent scenario reduction step and, equally important, revealing relationships between features of the uncertain parameters and the optimal decisions. Traditional scenario clustering methods (Beraldi and Bruni, 2014) assign scenarios with similar uncertainty realizations (i.e. uncertain parameter values) to the same cluster; we call this *realization-based* scenario clustering. The underlying assumption is that similar realizations lead to similar optimal decisions. However, closeness in uncertain parameter values may not necessarily result in closeness in decisions; rather it could be some other feature in the data that is a better predictor of the optimal decisions, but we do not know that feature a priori. In our case, we have the advantage that we already know the optimal recourse decisions y^*; hence, we can directly use that information to cluster the scenarios. We call this *recourse-based* scenario clustering, which can help the user *identify* the features in the uncertain parameters that best explain the optimal recourse decisions.

To perform the proposed recourse-based scenario clustering, we apply a standard k-means algorithm where given the full scenario set $\mathcal{S}$ and a desired number of clusters k, we aim to assign scenarios that exhibit the most similar recourse decisions to the same cluster. We obtain k sets of disjoint scenarios $\bar{\mathcal{S}} = \{\bar{\mathcal{S}}_1, \ldots, \bar{\mathcal{S}}_k\}$ with $\bar{\mathcal{S}}_i \subseteq \mathcal{S}$ for $i = 1, \ldots, k$ and $\mathcal{S} = \cup_{i=1}^{k} \bar{\mathcal{S}}_i$ by solving the following optimization problem:

$$\begin{aligned} \underset{\bar{\mathcal{S}}}{\text{minimize}} \quad & \sum_{i=1}^{k} \sum_{s \in \bar{\mathcal{S}}_i} \| y_s^* - \mu_i \| \\ \text{subject to} \quad & \mu_i = \frac{1}{|\bar{\mathcal{S}}_i|} \sum_{s \in \bar{\mathcal{S}}_i} y_s^* \quad \forall\, i = 1, \ldots, k, \end{aligned} \tag{SC$_k$}$$

where y_s^* are the optimal recourse decisions for scenario s and μ_i denotes the mean for scenario cluster i. Note that for the conventional realization-based scenario clustering, we would replace y_s^* in (SC$_k$) with (q_s, T_s, W_s, h_s).

2.2 Scenario reduction

Given k scenario clusters $\{\bar{\mathcal{S}}_1, \ldots, \bar{\mathcal{S}}_k\}$, we now want to find a representative scenario for each cluster such that the difference between the optimal first-stage solution to the resulting reduced SP problem, denoted by $\tilde{x}$, and the one to the original SP problem, x^*, is minimized. Ideally, we have $\tilde{x} = x^*$ as we would like to use the reduced model to explain the original solution, including elucidating why it is optimal.

The scenario reduction problem can be formulated as the following bilevel program:

$$\begin{aligned}
\underset{v,\tilde{x},\tilde{y}}{\text{minimize}} \quad & \|\tilde{x} - x^*\| && (\mathrm{SR}_k) \\
\text{subject to} \quad & (\tilde{x}, \tilde{y}) \in \underset{x,y}{\arg\min} \quad c^\top x + \sum_{i=1}^{k} \bar{p}_i \sum_{s \in \bar{\mathcal{S}}_i} q_s^\top y_s \\
& \qquad\qquad \text{subject to} \quad Ax \leq b \\
& \qquad\qquad\qquad \begin{bmatrix} v_s = 1 \\ T_s x + W_s y_s \leq h_s \\ y_s \geq 0 \end{bmatrix} \vee \begin{bmatrix} v_s = 0 \\ y_s = 0 \end{bmatrix} \quad \forall\, s \in \mathcal{S} \\
& \qquad\qquad\qquad x \geq 0 \\
& \sum_{s \in \bar{\mathcal{S}}_i} v_s = 1 \quad \forall\, i = 1, \ldots, k \\
& v \in \{0,1\}^{|\mathcal{S}|},
\end{aligned}$$

where the incorporation of the lower-level problem ensures that $\tilde{x}$ is optimal for the reduced SP model. The binary variable v_s equals 1 if scenario s is selected to be the representative scenario for the cluster it belongs to. The disjunctions in the lower-level problem indicate the selection of constraints and costs corresponding to the representative scenarios. The scenarios that are not selected, i.e., scenarios for which v_s equals 0, have their constraints relaxed and corresponding cost set to 0 (by setting $y_s = 0$). The disjunctions can be reformulated into mixed-integer linear constraints using standard reformulation techniques (Trespalacios and Grossmann, 2014). Lastly, $\bar{p}_i$ denotes the cumulative probability of cluster i, i.e., $\bar{p}_i = \sum_{s \in \bar{\mathcal{S}}_i} p_s$. Given a feasible v, the corresponding reduced SP model is (SP) with $\mathcal{S}$ replaced by $\widehat{\mathcal{S}} = \{s \in \mathcal{S} : v_s = 1\}$ and p_s replaced by $\hat{p}_s = \bar{p}_i$ where $s \in \bar{\mathcal{S}}_i$.

Remark 1 Note that even when the optimal first-stage decisions of the original and reduced SP models are the same, their optimal values may not. Normally, this does not hinder the interpretation of the solution to the reduced model. However, one should still check the difference in the optimal values since too large of a discrepancy may indicate an inadequate number of scenarios or weighting of the different scenarios in the reduced model.

3 Recourse reduction

The goal of recourse reduction is to obtain an SP formulation that has a significantly smaller set of recourse variables but still leads to the same optimal first-stage decisions x^* and the same or almost the same optimal value z^* as the original SP problem. The reduced set of recourse variables, which we refer to as the *principal recourse variables*, are chosen from the original set of recourse variables; the remaining variables are treated as first-stage variables in the reduced SP model. The principal recourse variables represent the decisions that benefit the most from the flexibility to change in the second stage depending on the realization of the uncertainty; as such, identifying those decisions immediately provides a valuable explanation of the SP solution. A smaller set of recourse decisions simplifies the analysis of the solution in general, and it also eases the implementation of the solution as the user has fewer decisions to make after observing the realizaed uncertainty.

3.1 Bilevel formulation

To formulate the recourse reduction problem, we introduce the set of indices of original recourse variables $\mathcal{R}$ and the binary variable v_r that equals 1 if y_r is chosen to be a recourse variable in the reduced SP problem. Given a selected set of recourse variables, encoded in the vector of binaries v, we can formulate the resulting reduced SP model as follows:

$$\begin{aligned} \underset{z,x,y}{\text{minimize}} \quad & z \\ \text{subject to} \quad & (z,x,y) \in \mathcal{F}(v) \end{aligned} \tag{RRSP(v)}$$

with the feasible region

$$\mathcal{F}(v) = \left\{ (z,x,y) : \begin{array}{l} z = c^\top x + \sum_{s \in \mathcal{S}} p_s q_s^\top y_s \\ Ax \le b \\ T_s x + W_s y_s \le h_s \quad \forall\, s \in \mathcal{S} \\ x \ge 0 \\ y_s \ge 0 \quad \forall\, s \in \mathcal{S} \\ y_{rs} - y_{r,s+1} \le M v_r \quad \forall\, r \in \mathcal{R}, s \in \mathcal{S} \setminus \{|\mathcal{S}|\} \\ y_{rs} - y_{r,s+1} \ge -M v_r \quad \forall\, r \in \mathcal{R}, s \in \mathcal{S} \setminus \{|\mathcal{S}|\} \end{array} \right\},$$

where all constraints from the full problem (SP) are considered as well as the cost function in the form of the first equation. In addition, the last two sets of constraints with the big-M parameter M represent non-anticipativity constraints that are enforced for each y_r for which $v_r = 0$. One can see that if $v_r = 0$, y_{rs} has to take the same value across all scenarios, which effectively renders y_r a first-stage variable. If $v_r = 1$, however, y_r remains a recourse variable as the value of y_{rs} can vary across different scenarios.

Given a desired number of recourse variables k, the recourse reduction problem can then be formulated as the following bilevel program:

$$\hat{z}_k = \min_{v,\tilde{z},\tilde{x},\tilde{y}} \quad \tilde{z} \tag{RR$_k$}$$

$$\text{subject to} \quad (\tilde{z},\tilde{x},\tilde{y}) \in \arg\min_{z,x,y} \{ z : (z,x,y) \in \mathcal{F}(v) \} \tag{1}$$

$$\tilde{x} = x^* \tag{2}$$

$$\sum_{r \in \mathcal{R}} v_r = k \tag{3}$$

$$v \in \{0,1\}^{|\mathcal{R}|}, \tag{4}$$

where we minimize the optimal value of the reduced SP problem $\tilde{z}$ such that it is as close as possible to z^*, the optimal value of (SP). The reduced SP model is embedded as a lower-level problem in constraint (1) that ensures that $(\tilde{z}, \tilde{x}, \tilde{y})$ is the optimal solution to (RRSP(v)). Equation (2) enforces that $\tilde{x}$ equals x^*, the optimal first-stage decisions from solving (SP). Equation (3) sets the number of selected recourse variables to k. Clearly, as we increase the number of recourse variables k, $\hat{z}_k$ will decrease such that $\hat{z}_0 \geq \hat{z}_1 \geq \cdots \geq \hat{z}_{|\mathcal{R}|} = z^*$. Hence, we can solve (RR$_k$) for different k's and choose a solution for which $\hat{z}_k$ is sufficiently close to z^*.

Remark 2 Equation (2) is an important constraint from an explainability standpoint. It ensures that x^*, the first-stage decisions that the user will implement, are also optimal for the reduced problem even if $\hat{z}_k \neq z^*$. This allows the user to use the reduced SP solution to investigate the relationship between first- and second-stage decisions and better understand why x^* is optimal. However, equation (2) is also what makes (RR$_k$) particularly difficult to solve, and (RR$_k$) may be infeasible for small k's due to this constraint.

3.2 Lower-bounding problem

The bilevel recourse reduction problem (RR$_k$) can be difficult to solve, especially when the original SP problem is large. In the following, we formulate a single-level lower-bounding problem, which is significantly more computationally efficient, and simple conditions under which it has the same optimal solution as (RR$_k$).

Proposition 1 *Consider the following problem:*

$$\begin{aligned} \underline{z}_k = \min_{v,z,x,y} \quad & z && \text{(LBP}_k\text{)} \\ \text{subject to} \quad & (z,x,y) \in \mathcal{F}(v) \\ & \sum_{r \in \mathcal{R}} v_r = k \\ & v \in \{0,1\}^{|\mathcal{R}|}, \end{aligned}$$

which is a (single-level) mixed-integer linear program (MILP). Its optimal solution provides a lower bound to (RR$_k$)*, i.e.* $\underline{z}_k \leq \hat{z}_k$.

Proof The following formulation is a relaxation of (RR$_k$) simply obtained by removing equation (2) from (RR$_k$):

$$\begin{aligned} \dot{z}_k = \min_{v,\tilde{z},\tilde{x},\tilde{y}} \quad & \tilde{z} \\ \text{subject to} \quad & (\tilde{z},\tilde{x},\tilde{y}) \in \arg\min_{z,x,y}\{z : (z,x,y) \in \mathcal{F}(v)\} \end{aligned}$$

$$\sum_{r\in\mathcal{R}} v_r = k$$
$$v \in \{0,1\}^{|\mathcal{R}|}.$$

Since it is a relaxation of (RR_k), we have $\dot{z}_k \leq \hat{z}_k$. In this bilevel program, no constraints are imposed in the upper level on variables in the lower level. In addition, the upper-level objective function is the optimal value of the lower-level problem. Hence, it can be further reformulated into:

$$\begin{aligned} \dot{z}_k \quad = \quad & \min_{v} \quad \min_{z,x,y}\{z : (z,x,y) \in \mathcal{F}(v)\} \\ & \text{subject to} \quad \sum_{r\in\mathcal{R}} v_r = k \\ & \qquad\qquad\quad v \in \{0,1\}^{|\mathcal{R}|}, \end{aligned}$$

which, after merging the two minimizations, leads to (LBP_k). Thus, we have $\underline{z}_k = \dot{z}_k \leq \hat{z}_k$.

In our computational experiments, we have observed that often, for sufficiently large k's, (RR_k) and (LBP_k) have the same solutions with regard to the choice of recourse variables. Sometimes this can be immediately verified, as stated in the following corollary.

Corollary 1 *Let $(\underline{v}, \underline{z}_k, \underline{x}, \underline{y})$ be the optimal solution to* (LBP_k)*. If $\underline{v}$ is also optimal for* (RR_k) *and the reduced SP problem* ($\mathrm{RRSP}(v)$) *with $v = \underline{v}$ has a unique optimal solution, then $\underline{x} = x^*$ and $\underline{z}_k = \hat{z}_k$.*

Proof If $\underline{v}$ is optimal for (RR_k), then x^* is an optimal solution to ($\mathrm{RRSP}(\underline{v})$) due to constraints (1) and (2). If x^* is the unique optimal solution to ($\mathrm{RRSP}(\underline{v})$), then $\underline{x} = x^*$ and $\underline{z}_k = \hat{z}_k$ since (LBP_k) reduces to ($\mathrm{RRSP}(\underline{v})$) for $v = \underline{v}$.

The uniqueness condition from Corollary 1 does not always hold. However, in the general case, we can solve another problem to check if a $\bar{v}$ that is optimal for (LBP_k) is also optimal for (RR_k).

Proposition 2 *Let $(\underline{v}, \underline{z}_k, \underline{x}, \underline{y})$ be the optimal solution to* (LBP_k)*. Solve the following verification problem:*

$$\begin{aligned} \bar{z}_k \quad = \quad & \min_{z,x,y} \quad z \\ & \text{subject to} \quad (z,x,y) \in \mathcal{F}(\underline{v}) \\ & \qquad\qquad\quad x = x^*. \end{aligned} \tag{VP_k}$$

If $\bar{z}_k = \underline{z}_k$, then $\underline{v}$ is also optimal for (RR_k) *and $\hat{z}_k = \underline{z}_k = \bar{z}_k$. Otherwise, $\bar{z}_k > \underline{z}_k$.*

Proof The purpose of solving (VP_k) is to check whether the first-stage decisions x^* are also optimal for ($\mathrm{RRSP}(\underline{v})$). This is true if $\bar{z}_k = \underline{z}_k$. In that case, $\underline{v}$ is feasible in (RR_k) and the resulting objective function value of (RR_k) is $\bar{z}_k$, which must also be the optimal value of (RR_k) since $\underline{z}_k \leq \hat{z}_k$ (from Proposition 1) and $\bar{z}_k = \underline{z}_k$. Hence, we have $\hat{z}_k = \underline{z}_k = \bar{z}_k$. Furthermore, since ($\mathrm{VP}_k$) is a restriction of ($\mathrm{LBP}_k$), we have $\bar{z}_k \geq \underline{z}_k$, and if x^* is not optimal for ($\mathrm{RRSP}(\underline{v})$), $\bar{z}_k > \underline{z}_k$.

Given Propositions 1 and 2, we could, instead of solving (RR_k), solve (LBP_k) for different k's and check if each solution is also optimal for (RR_k) by solving (VP_k), which is an LP. We may start with $k = 0$ and keep increasing k until we obtain a reduced set of recourse variables for which $\hat{z}_k = \underline{z}_k = \bar{z}_k$ and $\hat{z}_k$ is sufficiently close to z^*. Note that using this heuristic approach, we may terminate at a k that is larger than the one we would obtain from solving (RR_k). However, it can be very effective in cases where directly solving (RR_k) is computationally intractable.

4 Case studies

In this section, we present three case studies that demonstrate how the proposed scenario and recourse reduction techniques can aid in explaining SP solutions. First, we consider an illustrative supply chain planning problem where, despite the simplicity of the supply chain network, the solution to the original SP problem is difficult to analyze due to the large number of scenarios; here, we demonstrate the use of scenario reduction and the interpretation of the solution to the reduced model. The second case study addresses an electricity procurement scheduling problem, where we show how scenario clustering and reduction can be applied to obtain insights into the features of the uncertain parameters that affect the decisions the most. Finally, in the third case study, we use a larger instance of the supply chain planning problem to demonstrate the effectiveness of the proposed recourse reduction approach in reducing the complexity of the SP model. All problems were modeled using JuMP v1.1.1 (Dunning et al., 2017) in Julia v1.6.1 (Bezanson et al., 2017) and solved using Gurobi v9.5.2 (Gurobi Optimization, LLC, 2021). Bilevel problems specifically were modeled using BilevelJuMP v0.5.1 in Julia.

4.1 Supply chain planning with large number of demand scenarios

Consider a set of production plants that manufacture a certain product for which there are multiple customers with uncertain demands. In this problem, we assume that the production decisions at the plants need to be made before the demand uncertainty realizes; hence, these are our first-stage decisions. Once we know the demand, we can decide how much to transport from each plant to each customer and how much product we need to purchase externally to meet any shortfall in demand satisfaction; hence, these are our recourse decisions. This setting is, for example, typical for the industrial gas industry where production planning decisions are commonly made on a monthly basis while distribution decisions are made on a daily or weekly basis once the order book is finalized for that time period. This supply chain planning problem can be formulated as the following two-stage linear SP problem:

$$\underset{x,y,\bar{y}}{\text{minimize}} \quad \sum_{i\in\mathcal{I}} c_i x_i + \sum_{s\in\mathcal{S}} p_s \sum_{j\in\mathcal{J}} \left[\bar{q}_j \bar{y}_{js} + \sum_{i\in\mathcal{I}} q_{ij} y_{ijs}\right] \tag{5}$$

$$\text{subject to} \quad x_i \le C_i^{\max} \quad \forall\, i\in\mathcal{I} \tag{6}$$

$$\sum_{i \in \mathcal{I}} y_{ijs} + \bar{y}_{js} \geq \xi_{js} \quad \forall j \in \mathcal{J}, s \in \mathcal{S} \tag{7}$$

$$\sum_{j \in \mathcal{J}} y_{ijs} \leq x_i \quad \forall i \in \mathcal{I}, s \in \mathcal{S} \tag{8}$$

$$x_i \geq 0 \quad \forall i \in \mathcal{I} \tag{9}$$

$$y_{ijs} \geq 0 \quad \forall i \in \mathcal{I}, j \in \mathcal{J}, s \in \mathcal{S} \tag{10}$$

$$\bar{y}_{js} \geq 0 \quad \forall j \in \mathcal{J}, s \in \mathcal{S}, \tag{11}$$

where $\mathcal{I}$, $\mathcal{J}$, and $\mathcal{S}$ are the sets of plants, customers, and scenarios, respectively. The production quantity at plant i is denoted by x_i. The amount of product supplied from plant i to customer j in scenario s is captured by y_{ijs}. The amount of product that must be procured externally to meet the demand deficit at customer j in scenario s is denoted by $\bar{y}_{js}$. The unit production cost at plant i, unit transportation cost between plant i and customer j, and unit procurement cost at customer j are represented by coefficients c_i, q_{ij}, and $\bar{q}_j$, respectively. The demand for customer j in scenario s is ξ_{js}. The probability of scenario s is denoted by p_s. Constraints (6) limit the production at each plant i by the quantity $C_i^{\max}$. Constraints (7) ensure that the demand is satisfied at each customer, and constraints (8) ensure the product distribution from each plant does not exceed the production. Constraints (9)-(11) enforce the non-negativity on the decision variables. The objective function (5) represents the expected overall cost of operating the supply chain network.

In this case study, we consider a network of two plants (A and B) and three customers (1, 2, and 3), as shown in Fig. 2. The unit production costs at Plants A and B are \$100 and \$150, respectively, and maximum production amount at each plant, $C_i^{\max}$, is set to 38 units. The unit purchase cost at each customer is set to \$500. The unit transportation costs are as follows: \$90 for A→1, \$67.5 for A→2, \$90 for A→3, \$67.5 for B→1, \$30 for B→2, and \$67.5 for B→3. We consider 50 equally probable demand scenarios where the demand for each customer is sampled from $\mathcal{U}(0, 35)$.

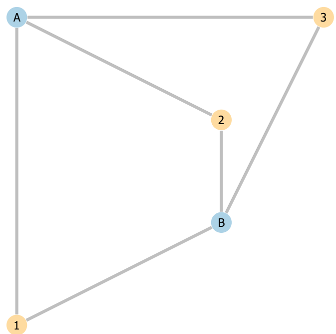


Fig. 2: A network of two plants and three customers denoted by blue and orange nodes, respectively. Edge lengths are proportional to the corresponding transportation costs.

With 50 scenarios, the solution to the SP model is difficult to interpret despite the rather simple network. Hence, we employ the proposed scenario reduction strategy to identify a simplified model that is easier to analyze. Setting $k = 5$, the recourse-based scenario clustering results in the five scenario clusters shown in Fig. 3. Here, we depict each scenario in the space of the three recourse variables y_{A1}, y_{A2}, and y_{A3}. For comparison purposes, we also performed conventional realization-based clustering and found that the resulting assignment has a within-cluster sum of squares in the recourse space, i.e. $\sum_{i=1}^{k} \sum_{s \in \bar{\mathcal{S}}_i} (y_s^* - \mu_i)^2$, that is about 51% greater than the one obtained from recourse-based clustering. This clearly indicates that scenarios that are similar in their uncertainty realizations may not lead to the same optimal recourse decisions.

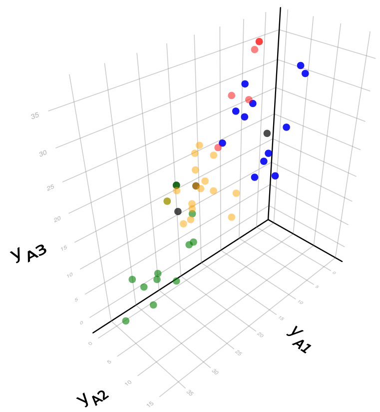


Fig. 3: Illustrating clusters obtained from recourse-based clustering for $k = 5$.

With the five scenario clusters, we solve scenario reduction problem and obtain the following five representative scenarios: Scenarios 4, 9, 14, 33, and 47 with probabilities 0.32, 0.24, 0.24, 0.12, and 0.08, respectively. The optimal first-stage decisions of the resulting reduced SP model are the same as the ones obtained from the original model with the full set of scenarios, namely 38 and 27 units of production from Plants A and B, respectively. Additionally, the optimal value of the reduced model differs from the one of the original model by less than 2%.

We analyze the reduced model's solution as a means of understanding the original model's solution. Fig. 4 shows the optimal decisions for all five scenarios of the reduced model. Here, the sizes of the plant and customer nodes represent the production amounts and product demands, respectively. The thickness of the edge between a plant and a customer indicates the amount of product transported. We can now make the following observations and draw some useful insights that help explain the SP solution:

1. We see that Plant A is at its production capacity while Plant B is not. This prompts us to compare the costs to serve each customer, which consist of

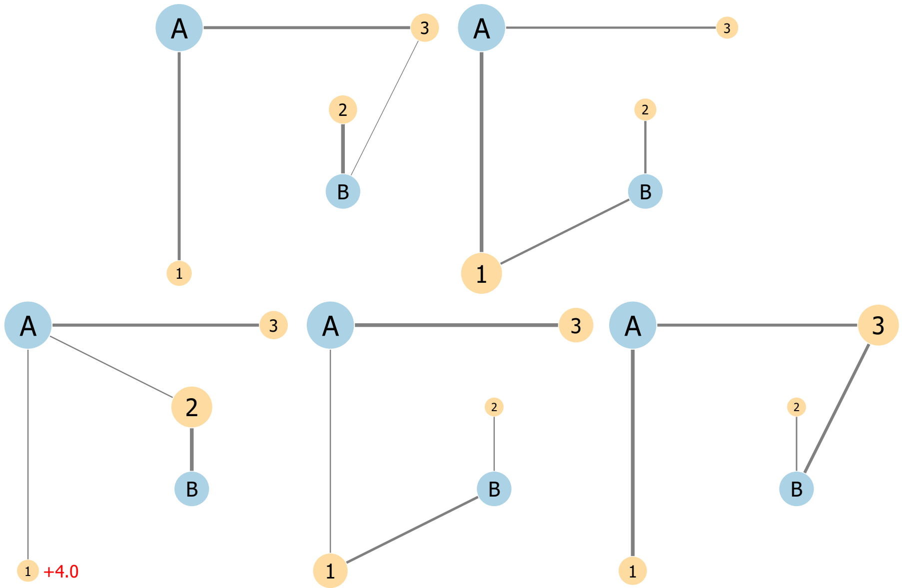


Fig. 4: Illustrating the first- and second-stage decisions obtained from solving the reduced model with five representative scenarios – 4 (top left), 9 (top right), 14 (bottom left), 33 (bottom center), and 47 (bottom right). The sizes of the plant and customer nodes correspond to the production amounts and product demands, respectively. The edge thickness corresponds to the amount transported between a plant and a customer. The quantity in red font ("+x") next to a customer node corresponds to the product purchased externally.

both production and transportation costs, from Plants A and B. Although, compared to Plant B, Plant A is located farther away from all customers, we find that the cost to serve any of the given customers is lower for Plant A due to its considerably lower production cost. This explains why the optimal solution suggests producing at Plant A as much as possible and use Plant B only to meet the demand that Plant A cannot.

2. In four out of the five scenarios, the capacities at both plants are sufficient to satisfy the demands of all customers, implying no external purchase is required in those scenarios. However, in Scenario 14, additional product is purchased externally, indicating that Plant B's production amount is not simply set to meet the largest demand among the five scenarios. This begs the question why it is optimal to produce this amount at Plant B.
3. In Scenario 14, Customer 2's demand exceeds Plant B's production capacity; thus, Plant A must pitch in to meet its demand despite being farther away. Also, the overall demand in the network exceeds the combined capacities of Plants A and B by 4 units, necessitating the external purchase of 4 units of the product at Customer 1. This results in a significantly higher second-stage cost compared to other scenarios, despite just slightly higher average demand per customer (see Fig. 5). Why do we not increase Plant B's production amount by 4 units to satisfy the demand of Customer 1 instead of purchasing the product externally, where the latter is certainly more expensive? Recall that the production rates are first-stage decisions; hence, the corresponding production

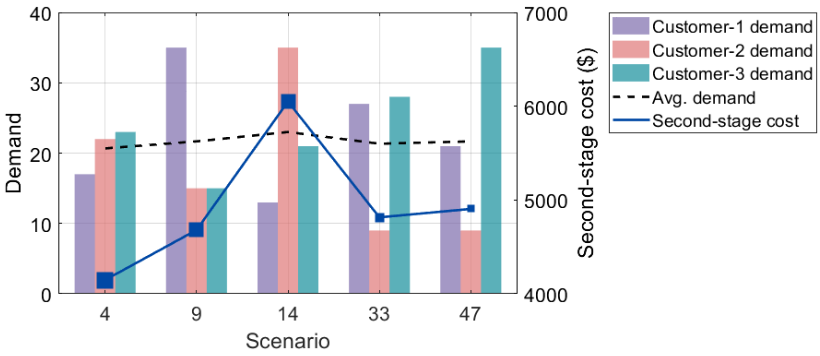


Fig. 5: Illustrating the chosen representative demand scenarios and the corresponding second-stage costs. The size of the markers is proportional to the representative probabilities of corresponding scenarios.

costs incur independent of the realization of the demand uncertainty. Transportation and purchase decisions are recourse decisions and differ across the five scenarios, so do the corresponding costs. In this case, since the 4 units of product purchase only occur in Scenario 14, which has a probability of 0.24, their contribution to the total expected cost is \$480. Increasing the capacity of Plant B by 4 units and transporting them to Customer 1 in Scenario 14, on the other hand, would increase the total expected cost by about \$665. So it is less expensive to purchase those 4 units of product externally.

4. Why should we not further reduce Plant B's production amount? Here, it helps to look at the total demands in all five scenarios, which are 62, 65, 69, 64, and 65 units, respectively. This means that if we produce less than 27 units at Plant B (in addition to the 38 units at Plant A), we will need to purchase additional product externally in Scenarios 9, 14, and 47. These three scenarios have a combined probability of 0.56; hence, purchasing one unit of product in these three scenarios would contribute \$280 to the total expected cost. This is in contrast to about \$179 for producing that one unit at Plant B and transporting it to Customers 1, 2, and 3 in Scenarios 9, 14, and 47, respectively, which explains why the optimal production amount at Plant B is 27 units.

The above analysis precisely explains the optimal first-stage decisions of the SP problem, which was relatively easy given the reduced set of scenarios. It would have been much more difficult if we tried to do so using the original full set of 50 scenarios.

4.2 Electricity procurement scheduling under price uncertainty

Electricity procurement is an important consideration in power-intensive industries. In operating such industrial plants, one typically needs to decide whether to purchase electricity from power contracts or from spot markets (Fig. 6). Power contract prices are relatively stable; however, the decision to purchase from them

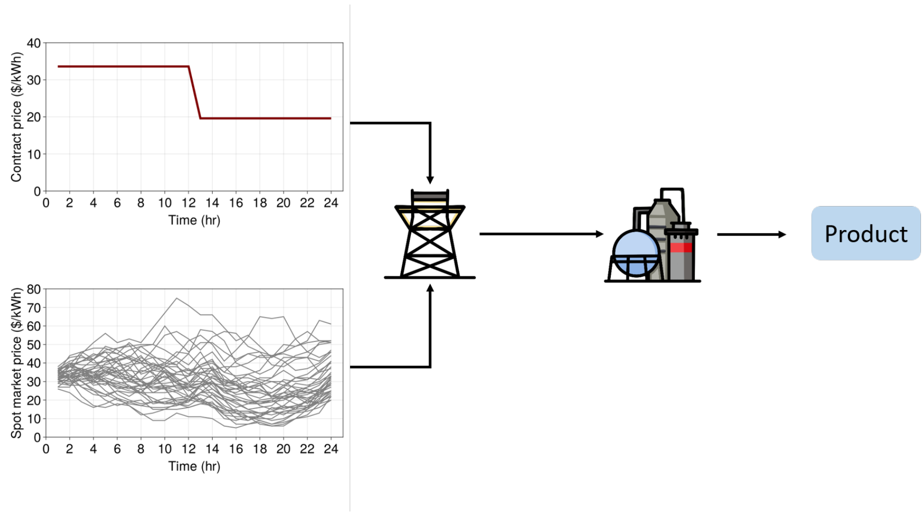


Fig. 6: Electricity can be purchased through the spot market or a contract. Contract prices are more stable but require ordering well ahead of time, whereas spot prices are more volatile but allow for short-notice purchases. We consider 40 price scenarios for the spot market over a 24-hour scheduling horizon.

needs to be made well ahead of time. On the other hand, spot electricity prices, in general, are highly time-sensitive and uncertain (may be lower or higher than the power contract price) and become known only shortly before the time of delivery. Hence, there is a trade-off to be optimized in purchasing electricity from power contracts and spot markets, which can be formulated as a two-stage SP problem. Here, we consider a simplified version of the model proposed by Zhang et al. (2016):

$$\underset{x,y,v}{\text{minimize}} \quad \sum_{t\in\mathcal{T}} c_t x_t + \sum_{s\in\mathcal{S}} p_s \sum_{t\in\mathcal{T}} \left[r_{ts} y_{ts} + a_t v_{ts} \right] \tag{12}$$

$$\text{subject to} \quad i^{\min} \le i^0 + \sum_{t'=1}^{t} (v_{t's} - d_{t'}) \le i^{\max} \quad \forall\, t \in \mathcal{T}, s \in \mathcal{S} \tag{13}$$

$$\sum_{t\in\mathcal{T}} (v_{ts} - d_t) \ge 0 \quad \forall\, s \in \mathcal{S} \tag{14}$$

$$m v_{ts} \le x_t + y_{ts} \quad \forall\, t \in \mathcal{T}, s \in \mathcal{S} \tag{15}$$

$$x_t \ge 0 \quad \forall\, t \in \mathcal{T} \tag{16}$$

$$y_{ts}, v_{ts} \ge 0, \quad \forall\, t \in \mathcal{T}, s \in \mathcal{S}, \tag{17}$$

where $\mathcal{T}$ and $\mathcal{S}$ denote the sets of time periods and spot price scenarios, respectively. The probability of occurrence of a scenario s is denoted by p_s. Electricity purchased from the power contract in time period t is denoted by x_t, whereas electricity purchased from the spot market and the production amount in time period t and scenario s are denoted by y_{ts} and v_{ts}, respectively. Note that x are the first-stage decisions while y and v constitute the recourse decisions. The unit cost of purchase from the power contract, cost of purchase from the spot market, and cost of production are represented by parameters c_t, r_{ts}, and a_t, respectively. The product demand in time period t and the initial inventory are denoted by d_t and i^0, respectively. Constraints (13) ensure that the inventory lies within the minimum ($i^{\min}$) and maximum ($i^{\max}$) permitted values at all times. Constraints (14) ensure the net production is at least as much as the overall demand during the scheduling horizon. The (minimum) electricity consumption in time period t and scenario s is assumed to be linearly dependent on the production amount defined by the coefficient m. The non-negativity bounds on the decision variables are enforced via constraints (16) and (17). The objective function captures the goal of minimizing the overall expected cost of purchasing electricity and production.

We consider a 24-hour scheduling horizon with hourly time discretization, minimum and maximum product inventory levels of 75 and 100 units, respectively, and an initial inventory of 50 units. The product demand in each time period is sampled randomly from $\mathcal{U}(28, 42)$. The contract and spot market electricity prices are shown in Fig. 6, while the unit production cost is set to \$3. We consider 40 equally probable scenarios for the spot prices, and the coefficient, m, defining the linear relationship between production and electricity consumption, is assumed to be 2.

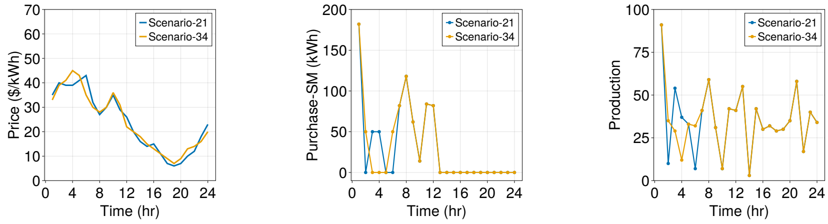
4.2.1 Scenario clustering

We want to use this example to demonstrate that often, with just simple scenario clustering, we can already gain valuable insights that help us explain the SP solution. To show the advantage of the proposed recourse-based scenario clustering method in this regard, we also compare the results with those obtained from conventional realization-based clustering.

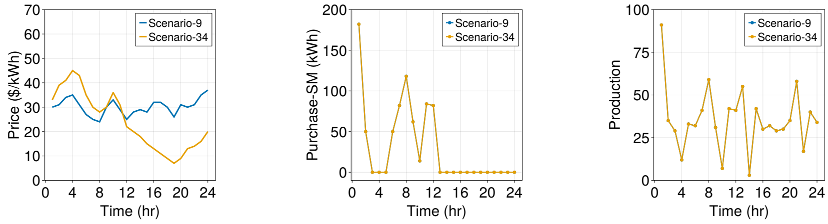
We start by setting the desired number of clusters, k, to 39. Since we have 40 scenarios, setting $k = 39$ returns 39 clusters where one of them consists of the two most similar scenarios. We perform both realization- and recourse-based clustering for which the two scenarios that are grouped into one cluster are shown in Fig. 7a and Fig. 7b, respectively. The figures show the corresponding spot price profiles along with the recourse decisions of purchasing electricity from the spot market and production. We see that the two scenarios found via realization-based clustering (Scenarios 21 and 34) are very similar in their spot price profiles, yet the associated recourse decisions differ significantly in time periods 2-6. In contrast, recourse-based clustering returns two scenarios (Scenarios 9 and 34) whose spot price profiles look quite different; however, they exhibit the exact same recourse decisions.

The above observation seems counter-intuitive at first, but after further investigation, we arrive at an explanation with the following two key insights:

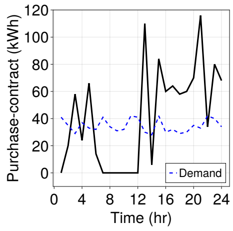
1. The spot price trend is a better predictor of the recourse decisions than the price magnitude. The reason is that for a fixed product demand, once the electricity purchase from the contract (first-stage decision) is finalized, we must purchase the remaining required electricity from the spot market, regardless of its price, to meet the product demand. Hence, the total amount of electricity purchased from the spot market is the same in all scenarios, and it is merely the timing of the electricity purchases that is optimized, which depends on how the spot price changes over time. This explains why the recourse decisions



(a) Realization-based clustering.



(b) Recourse-based clustering.



(c) Electricity purchase from contract.

Fig. 7: Illustrating two scenarios that get grouped together for the $k = 39$ case. Realization-based clustering reveals that scenarios that are very similar in magnitude do not necessarily have to have the same recourse decisions. In contrast, if there is a non-zero purchase from the spot market, then having a similar trend seems to play a more prominent role in having the same decisions across two scenarios.

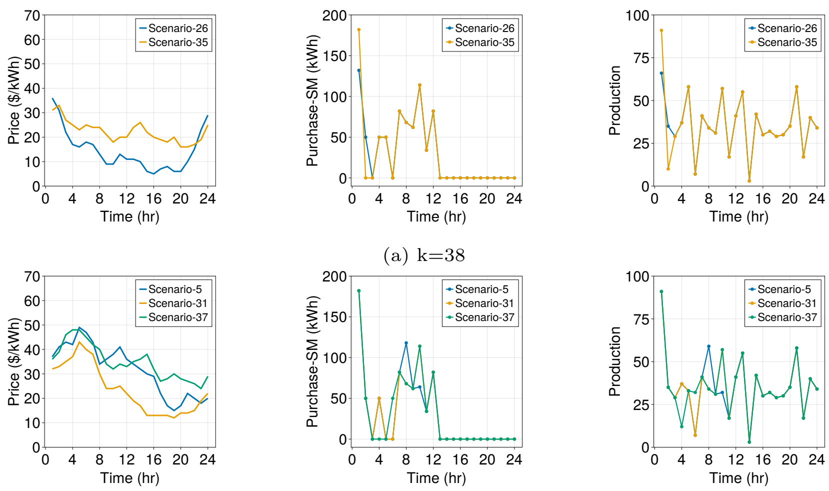
in Scenarios 9 and 34 are the same in time periods 1-12 as their price trends are the same despite the price values being quite different (see Fig. 7b). In Scenarios 21 and 34, on the other hand, the price profiles exhibit very similar magnitudes but often opposite trends in time periods 2-6 (see Fig. 7a); hence, the recourse decisions in those time periods are very different.

2. The price trends in Scenarios 9 and 34 are not always the same in time periods 13-24, yet the recourse decisions in those time periods are the same. A quick look at the first-stage decisions, as shown in Fig. 7c, reveals that the electricity purchased from the power contract in time periods 13-24 is enough to meet all electricity demand in that time frame so that no purchase from the spot market is needed. As a result, the trend in spot price in those time periods does not affect the recourse decisions.

To quantify and strengthen our reasoning behind our conclusions, we calculate the bivariate correlation coefficient, ρ, for every additional group of scenarios from $k = 39$ to $k = 35$, as shown in Table 1. Positive (similar trend) and negative (opposite trend) correlations are indicated by $\rho > 0$ and $\rho < 0$, respectively. In periods 1-24, for the most part, we see that the clustered scenarios are highly correlated, implying they follow a similar trend. The only ambiguous case is the one of $k = 36$, where the negative ρ for Scenarios 7 and 20 indicate a negative correlation despite being grouped together for having similar decisions. However, as we reasoned earlier that trends are immaterial in periods 13-24, it is more appropriate to calculate ρ only for periods 1-12 to avoid obtaining a misleading correlation. This correctly leads to $\rho > 0$, reiterating our observation of scenarios with similar trends leading to similar decisions. Finally, as shown in Fig. 8, this can also be observed in the grouped scenarios for cases $k = 38$ and $k = 37$.

Clusters (k)	**Clustered scenarios**	**Correlation coefficient (ρ)**	
		1-24 hr	**1-12 hr**
39	**(9,34)**	0.13	0.82
38	(9,34), **(26,35)**	0.73	0.94
37	(9,34), **(5,31,37)**	0.86	0.82
36	(9,34), (5,31,37), **(7,20)**	-0.42	0.36
35	(9,34), (5,31,37), (7,20), **(23,30)**	0.85	0.81

Table 1: Positive correlation between scenarios grouped together by recourse-based clustering indicates that similar price trends lead to similar decisions. Because trends are irrelevant during periods 13-24, ρ for periods 1-12 provides a more accurate representation of the correlation between scenarios. Mean ρ, i.e., the average of ρ between all possible pairs of scenarios, is reported for clusters with three scenarios.



(a) k=38

(b) k=37

Fig. 8: Illustrating scenarios grouped together via recourse-based clustering and the corresponding recourse decisions for cases $k = 38$ and $k = 37$. Again, we see very similar decisions, especially in time periods where the scenarios have similar trends, even if the scenarios differ considerably in magnitude.

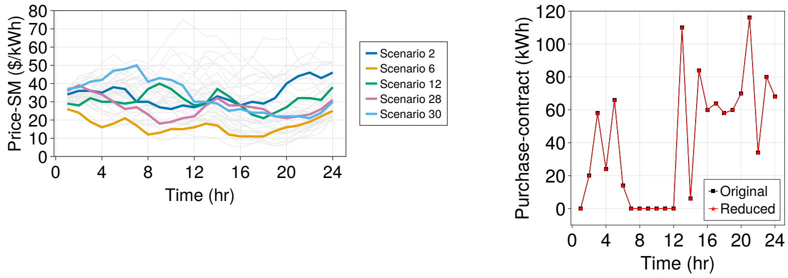


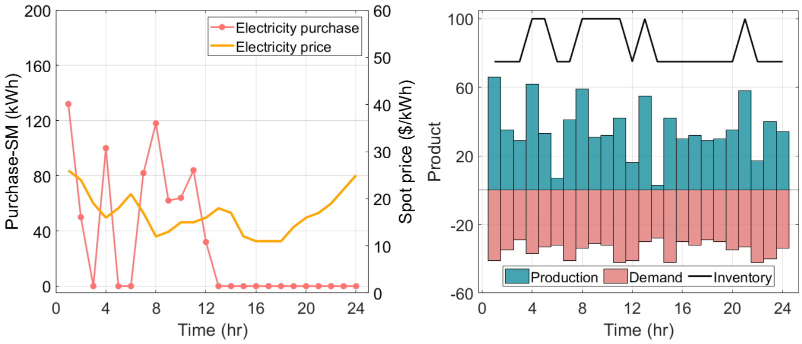
Fig. 9: Representative scenarios and first-stage decisions for $k = 5$.

4.2.2 Scenario reduction

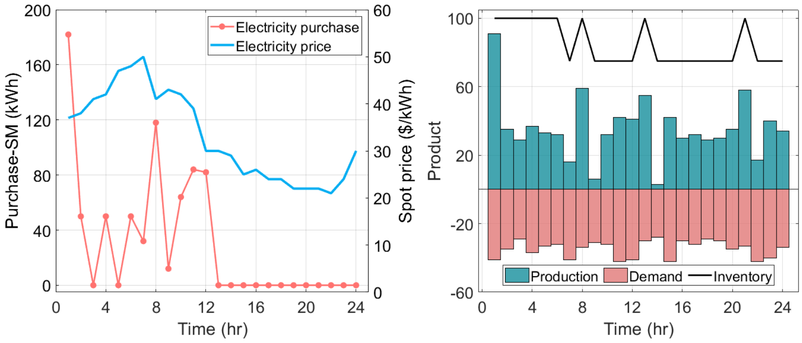
To analyze the overall SP solution, we apply the proposed scenario reduction approach to reduce the number of spot electricity price scenarios from 40 to 5. As depicted in Fig. 9, Scenarios 2, 6, 12, 28, and 30 are chosen as representative scenarios, with respective probabilities of 0.175, 0.1, 0.225, 0.225, and 0.275. Fig. 9 also shows that the optimal purchase amounts from the power contract, i.e. the first-stage decisions, are the same for the original and reduced models. In addition, we show, as examples, the second-stage solutions, i.e. the spot market purchase and production scheduling decisions, for Scenarios 6 and 30 in Fig. 10. Note that the subfigures on the right in Fig. 10 depict the production amounts as positive values while product demands are shown as negative numbers.

Now that we only have 5 scenarios to consider, we can more easily explain the first-stage decisions. Particularly, in time periods 13-24, the expected unit spot price lies in the range [$23.3/kWh, $34.3/kWh], which clearly exceeds the unit contract price of $19.6/kWh, resulting in only electricity purchases from the contract in the second half of the day. In contrast, in time periods 7-12, the expected unit spot price lies in [$25.8/kWh, $33.5/kWh], which is lower than the contract price of $33.6/kWh, making the spot market the preferred option for purchasing electricity in that time frame, as can be seen in Fig. 10.

The situation in time periods 1-6 is more complicated as we purchase electricity from both the contract and the spot market. Especially for the analysis here, it helps to be able to focus on a small number of scenarios. For example, from Fig. 10b, we can see that for Scenario 30, the spot electricity price is consistently higher than the contract price in time periods 1-6, yet we still purchase from the spot market. This, of course, can only be explained by realizing that the spot price in other scenarios, such as Scenario 6 (see Fig. 10a), is significantly lower. As a result, to minimize the overall expected cost, we should also purchase electricity from the contract in that time frame in order to balance hedging against the risk of high-price scenarios like Scenario 30 and allowing flexibility to take advantage of low-price scenarios like Scenario 6.



(a) Scenario 6.



(b) Scenario 30.

Fig. 10: Second-stage solutions for two of the five representative scenarios.

4.3 Supply chain planning with large number of recourse decisions

In this third case study, we consider a larger instance of the supply chain planning problem presented in Section 4.1, with 10 plants (A to J) and 15 customers. As shown in Fig. 11, the resulting network has 165 recourse decisions (150 transportation decisions plus 15 purchase decisions). The purchase cost at each customer node is set to \$125 per unit of product, and the maximum production limit at each plant is set to 12 units. Finally, we consider four demand scenarios with probabilities 0.05, 0.15, 0.30, and 0.50, and the demand for each customer is sampled from $\mathcal{U}(0, 12)$.

Solving the two-stage SP model corresponding to this large network results in decisions that are difficult to reason because of the complex network structure, which leads to a large number of recourse decisions. To alleviate this difficulty, we employ the recourse reduction method proposed in Section 3. Varying the desired number of recourse decisions k from 0 to 30, we solve both the lower-bounding problem (LBP$_k$) and the verification problem (VP$_k$). The results are plotted in Fig. 12, which shows the optimal values obtained from solving both problems as well as the optimal value of the original SP problem as a reference. Looking at the optimal values for (LBP$_k$), one can see that with only 18 recourse

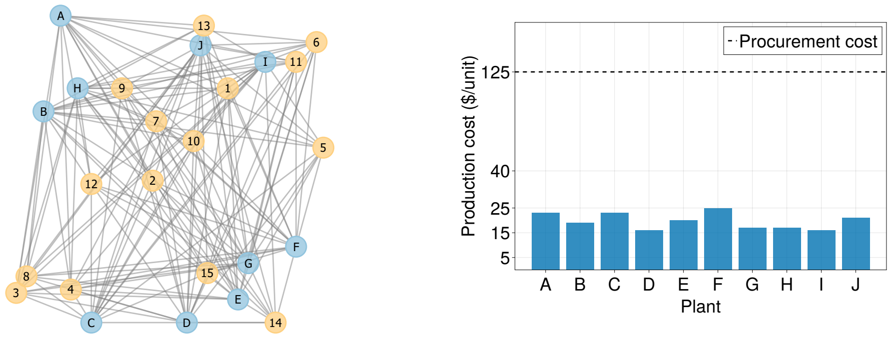


Fig. 11: Illustrating the original network consisting of 10 plants (A, B, ..., J) and 15 customers. The production cost at each plant and the procurement cost are shown on the right.

variables, we can already achieve the same optimal value as the original SP model. However, the solution to (VP_k) has a higher optimal value, which tells us that the selected 18 recourse variables do not yet result in the same optimal first-stage decisions. As we further increase k, we find that at $k = 23$, (LBP_k) and (VP_k) have the same optimal value, indicating that this is also the optimal solution to the bilevel program (RR_k). As such, we can now account for these 23 recourse variables, treat the remaining original recourse decisions as first-stage variables, and still ensure the same total expected cost and production decisions as with all 165 recourse variables. In the network depicted in Fig. 12, the 23 principal recourse decisions are highlighted in green, i.e. green edges and customer nodes represent the transportation and purchase decisions treated as recourse variables in the reduced model, respectively.

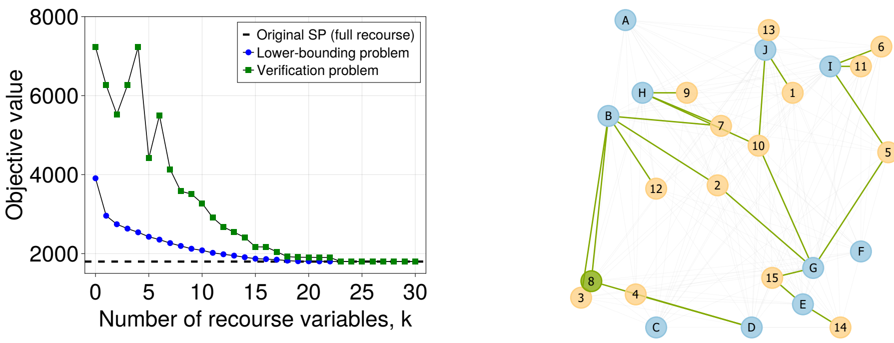


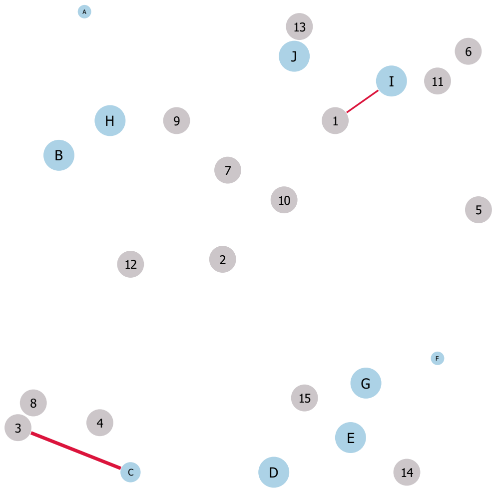
Fig. 12: On the left, optimal values obtained from solving the lower-bounding problem (LBP_k) and the verification problem (VP_k) for varying k's. On the right, a reduced network with only 14% (23 of 165) of the original recourse variables; selected recourse decisions are shown in green.

The solution to the reduced model with only the selected 23 recourse variables is shown in Fig. 13. The sizes of the plant and customer nodes (Figs. 13b-13e) indicate the production and demand at these nodes, respectively. The thickness of an edge between a plant and a customer is proportional to the transportation amount. The following are some key points regarding decisions made in the first stage: (i) As shown in Fig. 13a, some transportation decisions (red edges) are now made before the uncertainty in demand is realized. (ii) Plants A and F are chosen to not manufacture any product. The high production costs, combined with their relatively isolated locations (higher transportation costs), contribute to the decision to have no production at these plants.

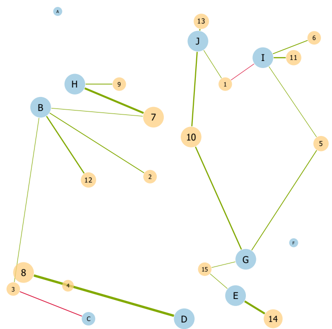
The user may now be interested in learning how product distribution varies with demand at a specific customer node and why the participating plants were selected. Now that we have a sparser network to analyze, it is much easier to answer such questions. We try to explain decisions involving Customer 8 as an example. In Scenario 1, the demand at Customer 8 is satisfied by Plant D. Plant C, despite being closer to Customer 8, is not chosen to meet its demand because it has only three units of production (due to the high production cost), all of which are used to meet the demand at Customer 3. In Scenario 2, we decide to buy 1 unit of product externally for Customer 8, which may be questionable due to the high purchase price. The reason for this is that Plants C and H, two other potential candidates to satisfy Customer 8's demand, are using all of their capacity to meet the demands of other nearby customers. Next, one might wonder why not externally purchase 1 unit of product for Customer 12 and transfer the 1 unit of product saved at Plant B to Customer 8. However, because the purchase price is the same for all customers, a more distant customer will always be preferred if a non-zero purchase decision must be made. Similarly, we can examine decisions involving any customer or plant of interest in any scenario. In summary, the reduction in recourse decisions significantly simplifies the analysis and hence the explainability of the SP solution.

5 Conclusion

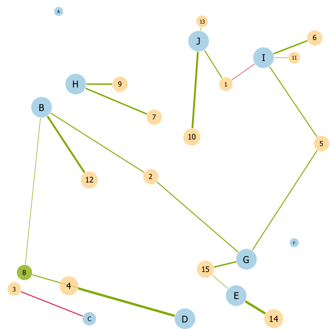
In this work, we addressed the challenge of low interpretability of decisions output by large SP models, which has for long hindered their use in real-world applications. We developed systematic approaches that enhance the explainability of a given SP solution by reducing the complexity of the model to an extent where it becomes easier to visualize the decisions and infer the reasons for making these decisions. The proposed techniques reduce the complexity of the problem along two dimensions – the number of scenarios and the number of recourse variables. For scenario reduction, our approach clusters scenarios based on the optimal recourse decisions and selects representative scenarios for the scenario clusters such that the optimal first-stage decisions of the resulting reduced SP model are close to the ones obtained from the original larger SP model. In recourse reduction, we identify a small set of principal recourse variables required to achieve the same optimal first-stage decisions while reaching the same or close to the same optimal value as the original SP model. The efficacy of the proposed explainability techniques was demonstrated in computational case studies involving problems of industrial relevance, namely supply chain planning and electricity procurement scheduling. We showed how the significant complexity reduction that could be obtained with min-



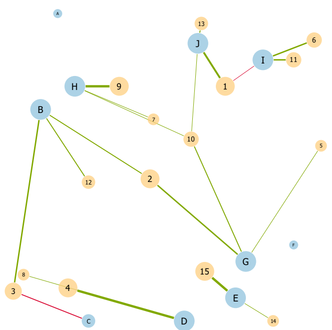
(a) First-stage decisions.



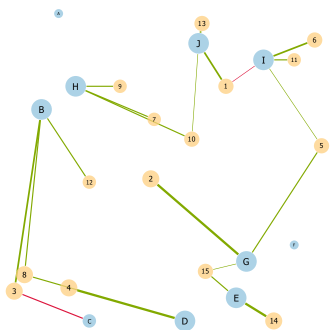
(b) Scenario 1.



(c) Scenario 2.



(d) Scenario 3.



(e) Scenario 4.

Fig. 13: Illustrating solution obtained by solving the reduced SP model with the 23 principal recourse decisions. Distribution or procurement decisions, which take the value zero, are omitted for brevity. Additionally, the thickness of an edge corresponds to the amount transferred from a plant to a customer.

imally dissimilar solutions ultimately made it much easier to explain the original SP solutions.

Acknowledgments

The authors gratefully acknowledge financial support from the National Science Foundation under Grant #2044077 and from Linde plc. R.G. further acknowledges financial support from a departmental fellowship sponsored by 3M and a Doctoral Dissertation Fellowship from the University of Minnesota.

References

Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115, 2020.

J Balasubramanian and IE Grossmann. Approximation to multistage stochastic optimization in multiperiod batch plant scheduling under demand uncertainty. *Industrial & engineering chemistry research*, 43(14):3695–3713, 2004.

Vaishak Belle and Ioannis Papantonis. Principles and practice of explainable machine learning. *Frontiers in Big Data*, page 39, 2021.

Patrizia Beraldi and Maria Elena Bruni. A clustering approach for scenario tree reduction: an application to a stochastic programming portfolio optimization problem. *Top*, 22:934–949, 2014.

Dimitris Bertsimas and Nishanth Mundru. Optimization-based scenario reduction for data-driven two-stage stochastic optimization. *Operations Research*, 2022.

Dimitris Bertsimas and Bartolomeo Stellato. The voice of optimization. *Machine Learning*, 110(2):249–277, 2021.

Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

John R Birge and Francois Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

Georgios L Bounitsis, Lazaros G Papageorgiou, and Vassilis M Charitopoulos. Data-driven scenario generation for two-stage stochastic programming. *Chemical Engineering Research and Design*, 187:206–224, 2022.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

Iain Dunning, Joey Huchette, and Miles Lubin. JuMP: A modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320, 2017.

Jitka Dupačová, Nicole Gröwe-Kuska, and Werner Römisch. Scenario reduction in stochastic programming. *Mathematical programming*, 95(3):493–511, 2003.

Yonghan Feng and Sarah M Ryan. Scenario construction and reduction applied to stochastic power generation expansion planning. *Computers & Operations Research*, 40(1):9–23, 2013.

Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2021.

Mike Hewitt, Janosch Ortmann, and Walter Rei. Decision-based scenario clustering for decision-making under uncertainty. *Annals of Operations Research*, 315 (2):747–771, 2022.

Kjetil Høyland and Stein W Wallace. Generating scenario trees for multistage decision problems. *Management science*, 47(2):295–307, 2001.

Sanjula Kammammettu and Zukui Li. Scenario reduction and scenario tree generation for stochastic programming using sinkhorn distance. *Computers & Chemical Engineering*, 170:108122, 2023.

Julien Keutchayan, Janosch Ortmann, and Walter Rei. Problem-driven scenario clustering in stochastic optimization. *arXiv preprint arXiv:2106.11717*, 2021.

Jesús Ma Latorre, Santiago Cerisola, and Andres Ramos. Clustering algorithms for scenario tree generation: Application to natural hydro inflows. *European Journal of Operational Research*, 181(3):1339–1353, 2007.

Zukui Li and Christodoulos A Floudas. Optimal scenario reduction framework based on distance of uncertainty distribution and output performance: I. single reduction via mixed integer linear optimization. *Computers & Chemical Engineering*, 70:50–66, 2014.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

Sergio Medina-Gonzalez, Ioannis Gkioulekas, Vivek Dua, and Lazaros G Papageorgiou. A graph theory approach for scenario aggregation for stochastic optimisation. *Computers & Chemical Engineering*, 137:106810, 2020.

Michael Patriksson, Ann-Brith Strömberg, and Adam Wojciechowski. The stochastic opportunistic replacement problem, part ii: a two-stage solution approach. *Annals of Operations Research*, 224(1):51–75, 2015.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1 (5):206–215, 2019.

Francisco Trespalacios and Ignacio E Grossmann. Review of mixed-integer nonlinear and generalized disjunctive programming methods. *Chemie Ingenieur Technik*, 86(7):991–1012, 2014.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

James E Ward and Richard E Wendell. Approaches to sensitivity analysis in linear programming. *Annals of Operations Research*, 27(1):3–38, 1990.

Qi Zhang, Jochen L Cremer, Ignacio E Grossmann, Arul Sundaramoorthy, and Jose M Pinto. Risk-based integrated production scheduling and electricity procurement for continuous power-intensive processes. *Computers & chemical engineering*, 86:90–105, 2016.

Weiguo Zhang and Xiaolei He. A new scenario reduction method based on higher-order moments. *INFORMS Journal on Computing*, 34(4):1903–1918, 2022.