Inverse Optimization with Discrete Decisions

Jerry Chua
Institute of Operations Research and Analytics, National University of Singapore
Institute for Infocomm Research (I2R), Agency for Science, Technology and Research (A*STAR)
jerry.chua@u.nus.edu

Ian Yihang Zhu
Department of Analytics and Operations, NUS Business School, National University of Singapore
ianyzhu@nus.edu.sg

Inverse optimization (IO) has emerged as a powerful framework for analyzing prescriptive model parameters that rationalize observed or prescribed decisions. Despite the prevalence of decision-making models with discrete variables, existing work has primarily focused on continuous and convex models, for which the corresponding IO problems are far easier to solve. This paper makes three contributions that broaden the foundations and applications of inverse optimization in discrete decision-making. First, we propose a new approach for approximating the inverse-feasible region of solutions to discrete optimization problems by modifying their linear relaxations. Second, we integrate these modified relaxations with existing methods to develop a unified framework for solving IO problems. Finally, through the lens of IO, we provide a fresh perspective on the sensitivity analysis of prescriptive solutions to discrete optimization problems, and demonstrate the computational advantages of our models and algorithms.

Key words: inverse optimization, discrete optimization, prescriptive analytics, linear relaxations, decomposition methods, sensitivity and post-optimality analysis

History: December 13, 2025.

1. Introduction

In recent years, inverse optimization (IO) has attracted significant attention in the operations research and prescriptive analytics communities (Chan et al. 2025). It has emerged as a powerful framework for analyzing prescriptive model parameters that explain and rationalize observed or prescribed decisions. While several modeling paradigms have been proposed, most existing work focuses on IO problems defined in their canonical form: given a feasible solution to a nominal optimization problem – known as the *forward problem* – the goal of the IO problem is to determine the objective coefficients of this forward problem under which the given solution is rendered optimal. When multiple objective coefficient vectors satisfy this condition, we seek the vector that optimizes a desired criterion while respecting any context-specific constraints on the coefficients.

IO problems defined in this canonical form can be found in a variety of contexts. For example, they appear widely in bilevel optimization and incentive design problems, where they are used to compute prices or incentives that render prescribed decisions optimal for decision-making agents

(e.g., Agarwal and Ergun 2010, Liu et al. 2024). Similarly, in data-driven and contextual optimization problems, they can be applied to infer latent utility coefficients that rationalize certain decisions (e.g., Lin et al. 2024), or to train machine learning models to make decision-focused predictions (e.g., Tang and Khalil 2024). These IO models have been applied across diverse sectors spanning energy, transportation, and healthcare systems; Chan et al. (2025) provide a comprehensive overview of such applications.

The defining feature of these IO problems is the *inverse-feasible region* (IFR), which describes the complete set of objective coefficients that render a given solution optimal to the forward problem (Tavaslıoğlu et al. 2018). By design, canonical IO problems are restricted to generate *inverse-feasible* vectors – that is, vectors lying in the inverse-feasible region – and the tractability of solving these problems thus depends on how easily this region can be represented. From this perspective, all IO solution methods can also be viewed through the way they represent or explore this region.

When the forward problem is a linear program, the inverse-feasible region can be reformulated using standard optimality conditions. In particular, objective coefficients that make a given solution optimal to the forward problem are precisely those that satisfy strong duality or complementary slackness conditions (Ahuja and Orlin 2001). In this context, IO problems are straightforward to solve, and the vast majority of IO applications have focused exclusively on forward problems that are linear programs (Chan et al. 2025).

When the forward problem contains discrete decision variables, the inverse-feasible region becomes much more difficult to handle, and the corresponding IO problems are significantly harder to solve. While exact reformulations of this region have been proposed – for example, through a super exponential set of subadditive functions (Schaefer 2009, Lamperski and Schaefer 2015) – they are largely conceptual in nature and difficult to implement in practice. Instead, cutting plane algorithms stand as the predominant approach for solving these IO problems, where conditions defining the inverse-feasible region are relaxed and cuts are iteratively generated to exclude candidate vectors that are not inverse-feasible (Wang 2009, Bodur et al. 2022).

While cutting plane algorithms have shown some promise, they suffer from a fundamental limitation: inverse-feasible vectors are, by definition, neither identified nor improved upon until the algorithm converges to an optimal solution to the IO problem. In practice, this means that regardless of whether the solution process takes minutes, hours, or days, no informative solutions are generated until the process terminates. This "all-or-nothing" approach can be highly limiting, most notably in settings where exact solution times may far exceed practical or reasonable time limits.

1.1. Summary of contributions

In this paper, we focus on solving canonical IO problems when the forward problem contains discrete decision variables. In this context, we make three major contributions. First, we propose

a novel approach for approximating the IFR by leveraging the linear relaxation of the forward problem. Second, we use these approximations to develop a generalized framework for solving IO problems, designed to iteratively generate and improve inverse-feasible vectors. Finally, we introduce novel applications of IO in the sensitivity and post-optimality analysis of discrete optimization problems, through which we derive insights into our methods and algorithms. A more detailed summary of each contribution is given below.

I. Tractable approximations of the inverse-feasible region. Mathematically, our approach is motivated by a rather simple question: for a discrete optimization problem, how can we best leverage its linear relaxation to approximate the inverse-feasible region? While linear relaxations have long served as a cornerstone of solution methods for discrete optimization, their role in inverse optimization has received surprisingly little attention. We demonstrate that typical linear relaxations may generate poor approximations of the IFR, but they can be restricted in a way that makes them significantly more effective for approximating this region. Specifically, we introduce the concept of *restricted linear relaxations* (RLRs), which are defined by adding a family of inequalities that are not valid for the linear relaxation. We demonstrate the advantages and versatility of RLRs through a number of theoretical results and practical examples.

II. Generalized solution methods for inverse optimization. We leverage RLRs to introduce a generalized framework for solving inverse discrete optimization problems. Specifically, we integrate RLRs into existing cutting plane approaches, which enables inverse-feasible solutions to be generated and refined iteratively. In this integrated approach, the RLR provides a unique and versatile mechanism for restricting the search space of IO solutions, allowing users to control the trade-off between pursuing small versus large solution improvements, and to specify termination conditions based on optimality gaps. Conceptually, this generalization can be viewed as a more complete primal-dual solution framework for inverse optimization.

III. New applications and computational insights. We broaden the scope of inverse optimization by demonstrating that a large family of *sensitivity and post-optimality analysis* (SPA) problems can be posed and solved as canonical IO problems. To the best of our knowledge, we are the first to bridge these two areas of research. By forming this connection, we show that our solution methods provide a unified framework for solving SPA problems in discrete optimization, covering both classic and modern SPA problems that we introduce for multi-objective and data-driven optimization. Using one such problem, we derive insights into the characteristics and performance of our solution methods. Our results demonstrate that under strict time limits, existing approaches can easily become irrelevant, while our generalized approach can still generate near-optimal solutions. We also investigate trade-offs between obtaining inverse-feasible solutions quickly and dedicating more time to achieving significantly better solutions.

The rest of our paper is organized as follows. Section 2 provides an overview of the relevant literature. In Section 3, we define the IO problem, present preliminary observations, and lay the foundations for subsequent developments. Section 4 introduces restricted linear relaxations and examines their use in approximating the IFR. Section 5 outlines the general framework for solving IO problems, and Section 6 focuses on the applications and numerical experiments. Section 7 concludes with directions for future research. All proofs can be found in the Electronic Companion.

1.2. Notation

We highlight common notation used in this paper. We use boldface uppercase letters for matrices (e.g., $\mathbf{A}$), boldface lowercase letters for vectors (e.g., $\boldsymbol{b}$), and calligraphic letters for sets (e.g., $\mathcal{X}$). The vector of all zeros (ones) is written as $\mathbf{0}$ ($\mathbf{1}$), and the i-th basis vector is given as $\boldsymbol{e}_i$. The sets $\mathbb{R}_+$ and $\mathbb{R}_-$ represent the positive and negative orthant, respectively. For a matrix $\mathbf{A}$, the i^{th} row vector is denoted as $\boldsymbol{a}_i \in \mathcal{A}$ and the i^{th} column vector is denoted as $\mathbf{A}_i \in \mathcal{A}$. For any set of vectors $\boldsymbol{a}_1, \cdots, \boldsymbol{a}_m$, its conic hull is given as $\text{cone}(\boldsymbol{a}_1, \cdots, \boldsymbol{a}_r) = \{\boldsymbol{x} \in \mathbb{R}^n | \boldsymbol{x} = \sum_{i=1}^r \lambda_i \boldsymbol{a}_i,\ \boldsymbol{\lambda} \in \mathbb{R}^r_+\}$. For a feasible region $\mathcal{X}$, we denote $\text{conv}(\mathcal{X})$ as its convex hull, and $\text{ext}(\mathcal{X})$ as the extreme points of this convex hull. Finally, for a polyhedral cone $\mathcal{K}$, we denote $\dim(\mathcal{K})$ as the dimensionality of the cone, and $\text{int}(\mathcal{K})$ to be the interior of this cone, if it exists.

2. Literature Review

We first review the relevant IO literature in Section 2.1, followed by the literature on sensitivity and post-optimality analysis in Section 2.2. These two areas have, to date, been largely examined independently; our review focuses mainly on work pertaining to discrete optimization. In Section 2.3, we briefly outline additional problems involving the IFR that may be of broader interest.

2.1. Inverse optimization

The IO literature has developed along two main streams of research that differ in whether the inverse-feasibility condition is imposed, that is, whether the output of the IO problem must render a given solution optimal to the forward problem (Chan et al. 2025). The first stream, which we refer to as canonical IO problems, enforces this condition, while the second stream, often termed as data-driven IO problems, relaxes this requirement. These two paradigms differ primarily in their intended applications. In this paper, we focus on canonical IO problems and review this body of work, while a brief discussion of data-driven IO is provided at the end of the subsection.

To date, most of the IO literature focuses on forward problems that are linear programs. For example, nascent literature focused on special cases of inverse linear programs for which one can derive analytical solutions based on the value of the optimal dual variables (e.g., Ahuja and Orlin 2001, Heuberger 2004). Similarly, many applications of inverse optimization focus on problems

where decision-making agents are assumed to be solving continuous and convex optimization problems (e.g., Marcotte et al. 2009, Agarwal and Ergun 2010). In such cases, strong duality and complementary slackness can be exploited to derive tractable reformulations of these IO problems.

Inverse discrete optimization problems, which are of more recent interest, are significantly harder to solve due of the difficulty of reformulating the inverse-feasible region (Schaefer 2009, Lamperski and Schaefer 2015, Bulut and Ralphs 2021). As mentioned in the introduction, cutting plane algorithms serve as the predominant approach for solving these problems (e.g., Wang 2009, Bodur et al. 2022), and we will show that they iteratively refine an outer approximation of the IFR. While this approach has shown some promise, it also possesses inherent limitations. Specifically, because the approach successively refines outer approximations of the IFR, no inverse-feasible solutions are found until the algorithm converges to the optimal solution. This results in an "all-or-nothing" solution process that is entirely uninformative until the algorithm converges, a process which often requires a significant amount of time. The only paper that has aimed to address this challenge is Duan and Wang (2011), which proposes a regularization-based heuristic designed to run in parallel with the cutting plane algorithm when solving a particular type of IO problem.

In this paper, we take a holistic approach to analyzing and solving inverse discrete optimization problems. We first propose a novel and versatile method for approximating the IFR using the linear relaxation of the discrete optimization problem. We then integrate these approximations into a unified solution framework for IO that overcomes the main limitations discussed above.

IO problems in their canonical form arise in diverse domains including incentive design, network pricing, decision-focused learning, cooperative games, and bilevel optimization; we refer to the survey paper of Chan et al. (2025) for a comprehensive overview. While most of these applications assume that agents solve convex optimization problems, they can be naturally extended to richer and more realistic environments where there are discrete decisions. As we will discuss in Section 2.2, our paper also broadens the scope of applications by showing that a large family of sensitivity and post-optimality analysis (SPA) problems can be formulated and solved as IO problems.

Finally, as noted earlier, there is also a growing body of literature on data-driven IO models (e.g., Bertsimas et al. 2015, Aswani et al. 2018, Mohajerin Esfahani et al. 2018, Shahmoradi and Lee 2022, Zattoni Scroccaro et al. 2025). These models are designed primarily for statistical and supervised learning, where the goal is to learn unknown parameters of a forward problem by minimizing suboptimality-based loss functions (Chan et al. 2025). Although this use case differs from the primary applications of canonical IO models, the latter may still be relevant, both for the sensitivity analysis of estimated parameters, and in situations where the canonical IO formulation appears as a subproblem within the parameter estimation process (e.g., Chan et al. 2024).

2.2. Sensitivity and post-optimality analysis

Inverse optimization is closely related to the domain of sensitivity and post-optimality analysis (SPA), although the two areas have evolved largely independently. For example, in the context of SPA, the inverse-feasible region is sometimes referred to as the sensitivity or stability region (e.g., Kılınç-Karzan et al. 2009, Andersen et al. 2023). For a given solution to a nominal optimization problem, many SPA problems aim to determine the ranges over which model parameters can vary without affecting the optimality of the given solution. Among these, the most widely studied is the *single-coefficient* SPA problem, which seeks to identify the largest interval within which a single objective coefficient can vary while all others are held fixed. The inverse-feasible region by definition contains the cartesian product of all such intervals.

SPA problems for linear programs are straightforward to solve, and the common example of the single-coefficient SPA problem is widely applied in practice and even taught in introductory optimization courses (Bertsimas and Tsitsiklis 1997). In contrast, SPA problems in discrete optimization are considerably more challenging, where existing methods can be highly problem-specific, analytically demanding, and/or lacking in generalizability. For example, to solve the single-coefficient SPA problem, early methods focus on techniques that leverage bounds on the objective value in each node of a pure branch-and-bound tree that we assume is used to generate a solution to the discrete optimization problem (Roodman 1974, Geoffrion and Nauss 1977, Jenkins 1982, Schrage and Wolsey 1985). Conversely, Klein and Holm (1979) shows that if the discrete optimization problem is instead solved using a pure cutting plane method, then the cuts that are binding at the optimal solution generates a valid region of objective coefficients for which this solution remains optimal. More recently, Kılınç-Karzan et al. (2009) focuses on computing sensitivity ranges for the objective coefficients of binary variables in a general discrete optimization problem. They show that a valid sensitivity range can be computed by taking the difference in objective values when the corresponding binary variable is forced to take the opposite value of the one that appears in the optimal solution. Andersen et al. (2023) proposes an alternative approach that reframes the sensitivity analysis problem as a bi-objective optimization problem, which is then solved using custom multi-objective optimization methods. Finally, attempts to approximate the IFR have also been proposed for specific problem structures. For example, Niendorf et al. (2015) and Niendorf and Girard (2017) examine the IFR for the traveling salesman problem and propose approximations based on graph-theoretic properties specific to their problem setting.

In this paper, we will show that a wide range of classic and modern SPA problems can be formulated as a canonical IO problem. This provides a unified methodological framework for analyzing the solutions to prescriptive models that contain discrete decision variables.

2.3. Additional applications of the IFR

The inverse-feasible region is also emerging in new applications that lie beyond the scope of the IO models considered in this paper. We include them below for broader interest, and as potential avenues for future research (see Section 7). For example, Wei and Zhang (2024) study conditions under which adjustable robust optimization problems yields the same objective value as its static counterpart, and show that the conditions depend on the IFR of a point in the polyhedral uncertainty set. Besbes et al. (2025) study contextual optimization problems where feedback is given in terms of optimal actions. Their optimal policy is based on approximations of the IFR of these actions. Chan et al. (2024) propose robust optimization models for recommending decisions using observations of past decisions. Their approach calibrates an uncertainty set on objective coefficients by requiring that this set intersects with IFRs associated with past decisions. Tang and Khalil (2024) and Berden et al. (2025) propose decision-focused methods for training models that predict the objective coefficients of linear optimization problems. Their approach aims to minimize deviations of these predictions from the IFR of optimal solutions. In the same context, Zhou et al. (2025) quantifies the risk associated with a prescriptive decision based on its inverse-feasible region.

3. Problem Definition and Preliminary Insights

We first begin by motivating and defining an inverse optimization problem in Section 3.1. We then outline preliminary ideas for approximating and solving these problems in Section 3.2. Section 3.3 provides a short summary of the main insights and presents a roadmap for the rest of the paper.

3.1. Problem definition and motivation

Consider a general discrete optimization problem of the form

$$\begin{aligned} \mathrm{FO}(\boldsymbol{\theta}, \mathcal{X}) \;:=\; \max_{\boldsymbol{x}} \quad & \boldsymbol{\theta}^\top \boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{x} \in \mathcal{X} := \Big\{ \boldsymbol{x} \;\Big|\; \mathbf{A}\boldsymbol{x} \le \boldsymbol{b},\; \boldsymbol{x} \in \mathbb{R}^{\ell} \times \mathbb{Z}^{n-\ell} \Big\}. \end{aligned} \tag{FO}$$

Throughout this paper, we refer to this nominal problem as the forward problem (FO), and to the set $\mathcal{X}$ as the forward-feasible region. We assume $\mathcal{X}$ is bounded, and denote its linear relaxation as $\mathcal{X}^{\mathrm{LR}} := \{\boldsymbol{x} \,|\, \boldsymbol{A}\boldsymbol{x} \le \boldsymbol{b},\, \boldsymbol{x} \in \mathbb{R}^n\}$. Then, given a solution $\hat{\boldsymbol{x}} \in \mathcal{X}$, the inverse-feasible region of $(\hat{\boldsymbol{x}}, \mathcal{X})$ is defined as the set of all possible vectors $\boldsymbol{\theta}$ under which $\hat{\boldsymbol{x}}$ is optimal in the forward problem, i.e.,

$$\mathcal{R}_{\mathrm{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) := \Big\{ \boldsymbol{\theta} \;\Big|\; \hat{\boldsymbol{x}} \in \arg\max \mathrm{FO}(\boldsymbol{\theta}, \mathcal{X}) \Big\}. \tag{R$_{\mathrm{INV}}$}$$

The following example provides a simple illustration of the forward and inverse-feasible region of a discrete optimization problem.

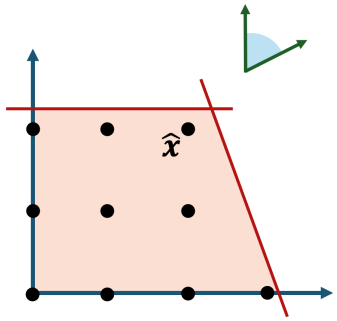


Figure 1 **A forward-feasible solution $\hat{x} \in \mathcal{X}$ and its corresponding inverse-feasible region $\mathcal{R}_{\text{inv}}(\hat{x}, \mathcal{X})$.**

Example 1. *Figure 1 depicts a forward-feasible region $\mathcal{X} := \{\boldsymbol{x} \in \mathbb{Z}^2 \,|\, x_1 + 0.4x_2 \leq 3.1, x_2 \leq 2.2\}$ and a feasible solution $\hat{\boldsymbol{x}} = (2, 2)$. The inverse-feasible region $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is illustrated by the blue cone in the top right corner, where any vector $\boldsymbol{\theta}$ drawn from this cone will render $\hat{\boldsymbol{x}}$ an optimal solution to the forward problem $\text{FO}(\boldsymbol{\theta}, \mathcal{X})$. In this figure, the shaded polyhedron represents the linear relaxation of $\mathcal{X}$, written as $\mathcal{X}^{\text{LR}} = \{\boldsymbol{x} \in \mathbb{R}^2 \,|\, x_1 + 0.4x_2 \leq 3.1, x_2 \leq 2.2\}$.*

We now define a general inverse optimization problem of the form

$$\begin{aligned} \text{IO}(\hat{\boldsymbol{x}}, \mathcal{X}) \;:=\; \min_{\boldsymbol{\theta}, \boldsymbol{s}} \quad & f(\boldsymbol{s}) \\ \text{s.t.} \quad & \boldsymbol{\theta} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) \\ & \boldsymbol{\theta} \in \boldsymbol{\Theta}(\boldsymbol{s}) \\ & \boldsymbol{s} \in \mathcal{S}. \end{aligned} \tag{IO}$$

In (IO), the main decision variables are $\boldsymbol{\theta}$, which correspond to the objective vector of the forward problem, and the first constraint ensures that $\boldsymbol{\theta}$ must lie in the inverse-feasible region $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$. Problem (IO) also contains auxiliary variables $\boldsymbol{s} \in \mathcal{S}$, where $\boldsymbol{\Theta}(\boldsymbol{s})$ defines a linear mapping $\mathcal{S} \to \mathbb{R}^n$ from $\boldsymbol{s}$ to $\boldsymbol{\theta}$, while $f(\boldsymbol{s})$ denotes the objective function to be optimized. Although these auxiliary variables are not required in all applications, they are included for generality and are particularly useful in the problems that we will examine; an example will be given shortly. Unless stated otherwise, we assume that $f(\boldsymbol{s})$ is a convex function and that $\mathcal{S}$ is a convex set.

Remark 1. *The formulation above generalizes IO models that do not have auxiliary variables, which can be expressed using $\boldsymbol{\Theta}(\boldsymbol{s}) := \boldsymbol{I}\boldsymbol{s}$, where $\boldsymbol{s} \in \mathbb{R}^n$ and $\boldsymbol{I}$ denotes the identity matrix.*

As discussed in Sections 1 and 2, problems in the form (IO) arise across diverse applications. In this paper, we broaden the scope of these applications by showing that (IO) can model and solve a wide range of sensitivity and post-optimality analysis (SPA) problems in discrete optimization. Specifically, SPA problems commonly aim to derive conditions under which a solution $\hat{\boldsymbol{x}} \in \mathcal{X}$ remains optimal when nominal model parameters are perturbed. The most commonly studied case is the single-coefficient SPA problem, which seeks to determine the largest interval within which a

single objective coefficient θ_i can deviate from its nominal value while the solution remains optimal to the forward problem (see Section 2.2). We show below that this is in fact an IO problem.

Observation 1. *Let $\hat{\boldsymbol{x}}$ denote an optimal solution to* $\mathrm{FO}(\hat{\boldsymbol{\theta}}, \mathcal{X})$ *under a nominal cost vector $\hat{\boldsymbol{\theta}}$. For a single coefficient $\hat{\theta}_i$, we can compute a maximal range $[\hat{\theta}_i + \Delta_i^-,\ \hat{\theta}_i + \Delta_i^+]$ for which $\hat{\boldsymbol{x}} \in \mathcal{X}$ remains optimal by solving a pair of inverse optimization problems, i.e.,*

$$\Delta_i^- = \min_{\boldsymbol{\theta}, s}\{s \,|\, \boldsymbol{\theta} = \hat{\boldsymbol{\theta}} + s\mathbf{e}_i,\ \boldsymbol{\theta} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})\} \tag{1a}$$

$$\Delta_i^+ = \max_{\boldsymbol{\theta}, s}\{s \,|\, \boldsymbol{\theta} = \hat{\boldsymbol{\theta}} + s\mathbf{e}_i,\ \boldsymbol{\theta} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})\}, \tag{1b}$$

where $\mathbf{e}_i$ denotes a unit vector where all elements $j \neq i$ are equal to zero.

In Section 6, we show that many other SPA problems can be specified using model (IO). To our knowledge, we are the first to bridge these two areas of research, and this connection will facilitate a unified IO-based framework for solving both existing and new SPA problems.

3.2. IFR approximations using surrogates

The defining feature of IO problems is the inverse-feasible region $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$. In this section, we outline a conceptual framework for generating approximations of this region using the idea of *surrogates* for $\mathcal{X}$. We rely on a simple but important observation, which is that $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is a polyhedral cone that can be represented through a set of supporting hyperplanes.

Observation 2. *The inverse-feasible region $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is equivalent to the polyhedral cone*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}) := \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top(\hat{\boldsymbol{x}} - \boldsymbol{x}) \geq 0,\ \forall \boldsymbol{x} \in \mathcal{X}\right\}. \tag{2}$$

Equation (2) reformulates the optimality condition in ($\mathrm{R_{INV}}$) explicitly using a set of hyperplanes which ensure that $\hat{\boldsymbol{x}}$ attains an objective value no smaller than that of any other solution $\boldsymbol{x} \in \mathcal{X}$. In general, this exact representation cannot be used directly, as it is based on an exponentially large or semi-infinite set of hyperplanes, each defined by a feasible solution of $\mathcal{X}$ that must first be generated. Nonetheless, equation (2) provides a valuable perspective for approximating the IFR.

Observation 3. *We can approximate $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ with $\mathcal{K}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}})$ where $\tilde{\mathcal{X}}$ defines a surrogate for $\mathcal{X}$.*

Observation 3 sets the stage for the rest of the paper, through which we develop new surrogates for $\mathcal{X}$. We start by presenting two simple surrogates that we will build upon later.

Remark 2. *From this point onward, we use the notation $\mathcal{K}(\hat{\boldsymbol{x}}, .)$ to refer to approximations of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$. This choice is deliberate: for a given surrogate $\tilde{\mathcal{X}}$, the sets $\mathcal{K}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}})$ and $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}})$ coincide when $\hat{\boldsymbol{x}} \in \tilde{\mathcal{X}}$, but differ when $\hat{\boldsymbol{x}} \notin \tilde{\mathcal{X}}$, since $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}}) = \emptyset$ whereas $\mathcal{K}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}}) \neq \emptyset$.*

3.2.1. Surrogate I: subsets of $\mathcal{X}$. A natural example of a surrogate for $\mathcal{X}$ is $\hat{\mathcal{X}} \subseteq \mathcal{X}$, where $\hat{\mathcal{X}}$ defines an explicit list of solutions known to lie in $\mathcal{X}$. By definition, $\mathcal{K}(\hat{\boldsymbol{x}}, \hat{\mathcal{X}})$ forms an outer approximation of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, since $\mathcal{K}(\hat{\boldsymbol{x}}, \hat{\mathcal{X}})$ is defined by a subset of the hyperplanes from (2). For any $\hat{\mathcal{X}}^2 \subseteq \hat{\mathcal{X}}^1 \subseteq \mathcal{X}$, it must also be true that $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \hat{\mathcal{X}}^1) \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \hat{\mathcal{X}}^2)$. This implies that enumerating more feasible solutions in $\mathcal{X}$ can yield tighter outer approximations of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$.

Existing cutting plane (CP) algorithms for inverse optimization operate precisely through this mechanism. Specifically, the standard CP approach alternates between a master problem that solves a relaxed version of (IO) where $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is replaced with $\mathcal{K}(\hat{\boldsymbol{x}}, \hat{\mathcal{X}})$ for some $\hat{\mathcal{X}} \subset \mathcal{X}$, i.e.,

$$\begin{aligned} \min_{\boldsymbol{\theta}, \boldsymbol{s}} \quad & f(\boldsymbol{s}) \\ \text{s.t.} \quad & \boldsymbol{\theta} \in \mathcal{K}(\hat{\boldsymbol{x}}, \hat{\mathcal{X}}) \\ & \boldsymbol{\theta} \in \boldsymbol{\Theta}(\boldsymbol{s}) \\ & \boldsymbol{s} \in \mathcal{S}, \end{aligned} \tag{3}$$

and a subproblem that adds new solutions $\boldsymbol{x} \in \mathcal{X}$ to a growing set $\hat{\mathcal{X}} \subset \mathcal{X}$ (Bodur et al. 2022). These algorithms will eventually generate the optimal IO solution because $\text{ext}(\mathcal{X}) \subseteq \mathcal{X}$, which defines the set of extreme points of the convex hull of $\mathcal{X}$, forms an exact surrogate for $\mathcal{X}$ (i.e., $\mathcal{K}(\hat{\boldsymbol{x}}, \text{ext}(\mathcal{X})) = \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X})$); in the worst case, the algorithm will terminate after enumerating all points in $\text{ext}(\mathcal{X})$. However, because $\mathcal{K}(\hat{\boldsymbol{x}}, \hat{\mathcal{X}})$ provides only an outer approximation of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, the solutions generated by (3) are not feasible to (IO) until $\hat{\mathcal{X}}$ becomes rich enough to solve the IO problem. This is a process that can take an arbitrarily long time; see Sections 1 and 2.

3.2.2. Surrogate II: the linear relaxation of $\mathcal{X}$. Our paper is motivated by how we can best use the linear relaxation of $\mathcal{X}$, denoted as $\mathcal{X}^{\text{LR}}$. Note that since $\mathcal{X} \subseteq \mathcal{X}^{\text{LR}}$, $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ contains a superset of the hyperplanes that define $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X})$, which implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is an inner approximation of the inverse-feasible region. Furthermore, since $\mathcal{X}^{\text{LR}}$ is a convex polyhedral set, we can derive a simple polyhedral reformulation of the cone $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$, as shown below.

Observation 4. *Given $\mathcal{X}^{\text{LR}} := \{\boldsymbol{x} \in \mathbb{R}^n \mid \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}\}$, we can reformulate $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ as*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) = \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{p}^\top(\mathbf{A}\hat{\boldsymbol{x}} - \boldsymbol{b}) = 0,\ \boldsymbol{p}^\top\mathbf{A} = \boldsymbol{\theta},\ \boldsymbol{p} \geq \mathbf{0}\right\}. \tag{4}$$

Note that if (FO) is a linear program, meaning that $\mathcal{X} = \mathcal{X}^{\text{LR}}$, then $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is equivalent to (4).

Equation (4) is derived using standard complementary slackness conditions. Note that for any index i, $p_i = 0$ if $\boldsymbol{a}_i^\top\hat{\boldsymbol{x}} - b_i \neq 0$, and $p_i \geq 0$ if $\boldsymbol{a}_i^\top\hat{\boldsymbol{x}} - b_i = 0$. This implies that the set $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ is defined as the conic hull of normal vectors $\boldsymbol{a}_i$ that correspond to binding constraints at $\hat{\boldsymbol{x}}$, i.e.,

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) = \text{cone}(\{\boldsymbol{a}_i\}_{i \in \mathcal{I}}), \tag{5}$$

where $\mathcal{I}$ denotes the indices of all constraints where $\boldsymbol{a}_i^\top\hat{\boldsymbol{x}} = b_i$.

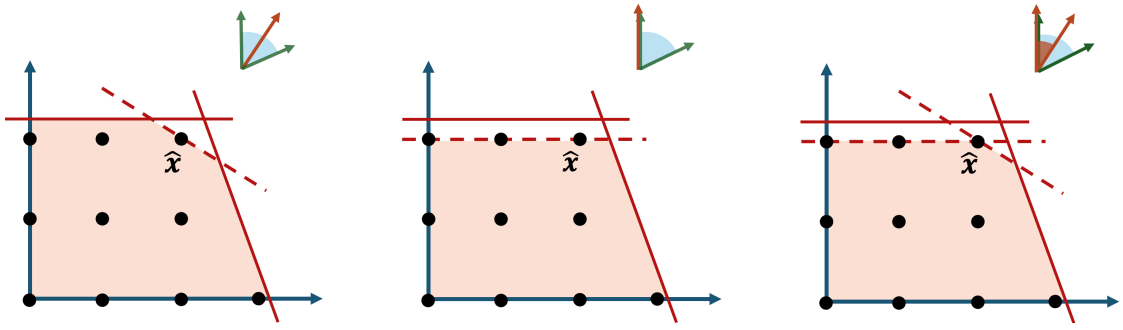
Observation 4 implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ is tractable and provides a straightforward inner approximation of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$. Using $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$, we can then consider the following approximation of (IO),

$$\begin{array}{ll} \min\limits_{\boldsymbol{\theta}, \boldsymbol{s}} & f(\boldsymbol{s}) \\ \text{s.t.} & \boldsymbol{\theta} \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \\ & \boldsymbol{\theta} \in \boldsymbol{\Theta}(\boldsymbol{s}) \\ & \boldsymbol{s} \in \mathcal{S}, \end{array} \tag{6}$$

which if feasible, would generate a feasible solution to (IO).

While this IFR approximation is simple and tractable, its main limitation is that the solution generated by (6) may not be very good. For example, Figure 1 shows an extreme case where $\hat{\boldsymbol{x}}$ lies strictly in the interior of $\mathcal{X}^{\text{LR}}$, implying that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) = \{\boldsymbol{0}\}$. More generally, the quality of this IFR approximation depends entirely on the tightness of the linear relaxation at $\hat{\boldsymbol{x}}$.

Adding new valid inequalities to the linear relaxation $\mathcal{X}^{\text{LR}}$ can potentially improve the resulting IFR approximation (recall that a *valid inequality* is any inequality $\boldsymbol{\alpha}^\top \boldsymbol{x} \leq \beta$ that holds for all $\boldsymbol{x} \in \mathcal{X}$ (Cornuéjols 2008)). Specifically, equation (5) implies that new valid inequalities would only improve the resulting IFR approximation if they are both binding at $\hat{\boldsymbol{x}}$ and strengthen $\mathcal{X}^{\text{LR}}$. Figure 2 illustrates the addition of such valid inequalities to the linear relaxation of Figure 1, and the improved IFR approximations are shown in the top right corner of each subfigure. Nonetheless, the task of repeatedly generating such valid inequalities is far from straightforward (e.g., Chvátal et al. 2013); any procedure for doing so would be conceptually equivalent to directly identifying inverse-feasible vectors $\boldsymbol{\alpha} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ that lie outside of $\text{cone}(\{\boldsymbol{a}_i\}_{i \in \mathcal{I}})$.



(a) (b) (c)

Figure 2 **Tighter linear relaxations and their corresponding IFR approximations. In Figure 2a and 2b, the linear relaxation shown in Figure 1 is improved by adding a valid inequality, and each IFR approximation consists of a single (long orange) vector. Figure 2c illustrates when both inequalities are combined, leading to an IFR approximation defined by the cone between the two (long orange) vectors.**

SURROGATES FOR $\mathcal{X}$	APPROXIMATIONS OF $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$	USE IN SOLVING IO PROBLEMS
$\hat{\mathcal{X}} \subseteq \mathcal{X}$	outer approximation	CP Algorithms [iteratively increases size of $\hat{\mathcal{X}}$] (Wang 2009, Bodur et al. 2022)
$\mathcal{X}^{\text{LR}} \supseteq \mathcal{X}$	inner approximation	N/A (see Section 3.2.2)
$\mathcal{X}^{\text{LR}}(\mathcal{W})$ (**RLRs**)	Pure-RLR: inner approx.	**RLR-CP Algorithms** iteratively reduces size of $\mathcal{X}^{\text{LR}}(\mathcal{W})$ while increasing size of $\hat{\mathcal{X}} \subseteq \mathcal{X}$
	Mixed-RLR: "mixed" approx.	

Table 1 **An overview of how different surrogates for $\mathcal{X}$ can lead to different approximations of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, as well as how these surrogates can be used to solve inverse optimization problems.**

3.3. Summary and roadmap

The main challenge in solving (IO) lies in the difficulty of handling the inverse-feasible region. In Section 3.2, we introduced two simple approximation strategies for $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ – or equivalently for $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X})$ – by replacing $\mathcal{X}$ with a surrogate $\tilde{\mathcal{X}}$. When this surrogate is defined as a subset of forward-feasible solutions ($\tilde{\mathcal{X}} \subseteq \mathcal{X}$), problem (IO) can be solved iteratively using a cutting plane algorithm, but at the cost of long solution times and a solution process that is entirely uninformative until it terminates. On the other hand, defining the surrogate as the linear relaxation $\mathcal{X}^{\text{LR}}$ yields an approximation of (IO) that is much easier to solve, but may produce low-quality solutions with no systematic means of improving these solutions.

Table 1 summarizes the main ideas discussed thus far and offers a roadmap for the material that follows. Specifically, in Section 4, we introduce *restricted linear relaxations* (RLRs) as a new class of surrogates for $\mathcal{X}$. RLRs are constructed by leveraging existing valid inequalities in a manner that yields substantially richer IFR approximations. In Section 5, we then use RLRs to develop a generalized framework for solving IO problems, in which they serve as a bridge between cutting plane algorithms and linear relaxations. This framework mitigates the inherent limitations of each approach while preserving their respective strengths.

Remark 3. *Going forward, we assume that $\hat{\boldsymbol{x}}$ lies on the boundary of $\mathcal{X}^{\text{LR}}$, or equivalently, that we know of at least one valid inequality (i.e., constraint) that is binding at $\hat{\boldsymbol{x}}$. This assumption is justified for two reasons. First, in most real-world problems, $\hat{\boldsymbol{x}}$ will naturally bind at a large number of constraints, such as equality constraints present in (FO), or at variable bounds (e.g., if $\mathcal{X}$ contains even a single binary variable $x_i \in \{0,1\}$, then $x_i \geq 0$ or $x_i \leq 1$ would be an example of*

such a valid inequality). Second, in SPA problems, $\hat{\boldsymbol{x}}$ is the optimal solution to a known, nominal forward problem $\mathrm{FO}(\hat{\boldsymbol{\theta}}, \mathcal{X})$. *By definition, $\hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}}$ would define such a valid inequality.*

4. IFR Approximations using Restricted Linear Relaxations

In this section, we introduce RLRs as a new class of surrogates for generating approximations of the inverse-feasible region. Loosely speaking, RLRs are defined by taking specific valid inequalities and modifying their right-hand-side values so that they are no longer valid. We show that these seemingly simple modifications yield a versatile approximation framework with desirable properties.

4.1. Definitions

We first present a general definition for restricted linear relaxations, before further classifying them into two types. An illustrative example is provided afterwards.

4.1.1. General definition of an RLR. We begin by introducing the term *restricting inequality*, which is defined with respect to $\hat{\boldsymbol{x}} \in \mathcal{X}$. Specifically, given an inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$, we say it is a restricting inequality if $\boldsymbol{w}^\top \hat{\boldsymbol{x}} > \tau$ and $\boldsymbol{w} \in \mathcal{R}_{\mathrm{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$.

Definition 1. *Given a linear relaxation $\mathcal{X}^{\mathrm{LR}}$, a restricted linear relaxation $\mathcal{X}^{\mathrm{LR}}(\mathcal{W})$ is defined as*

$$\mathcal{X}^{\mathrm{LR}}(\mathcal{W}) := \{\boldsymbol{x} \in \mathcal{X}^{\mathrm{LR}} \,|\, \boldsymbol{x} \in \mathcal{W}\} \tag{7}$$

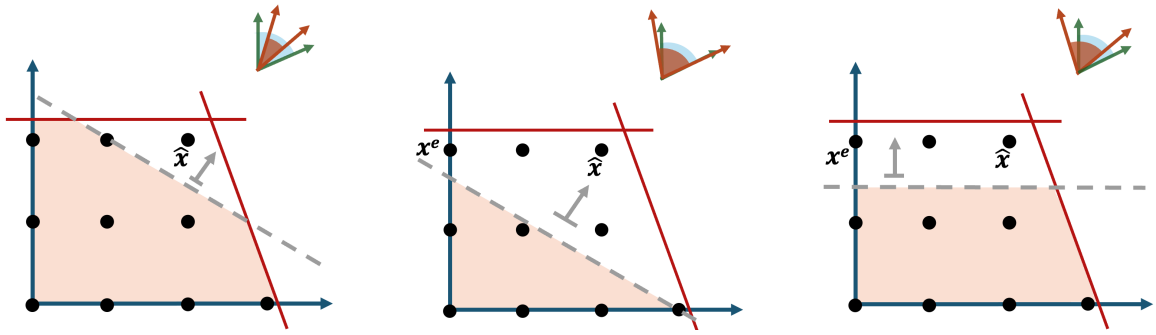
where $\mathcal{W} := \{\boldsymbol{x} \in \mathbb{R}^n \,|\, \mathbf{W}\boldsymbol{x} \leq \boldsymbol{\tau}\}$ is composed of only restricting inequalities for $(\hat{\boldsymbol{x}}, \mathcal{X})$.

By definition, the set of restricting inequalities in $\mathcal{W}$ are not valid for the linear relaxation $\mathcal{X}^{\mathrm{LR}}$, as they exclude the feasible solution $\hat{\boldsymbol{x}} \in \mathcal{X}$. Nonetheless, they are closely related to valid inequalities for $\mathcal{X}^{\mathrm{LR}}$. Specifically, given any valid inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \bar{t}$ that is tight at $\hat{\boldsymbol{x}}$, meaning that $\bar{t} = \boldsymbol{w}^\top \hat{\boldsymbol{x}}$, we can define a restricting inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$ simply by setting $\tau < \bar{t}$. For this reason, we refer to $\eta := \bar{t} - \tau \geq 0$ as the *depth* of the restricting inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$.

4.1.2. Pure and mixed-RLRs. To better describe their mathematical properties, we classify RLRs into two types by distinguishing between whether the restricting inequalities satisfy an additional condition. Specifically, given a $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$, we say it is a *separating* restricting inequality if it satisfies the additional condition that

$$\boldsymbol{w}^\top \boldsymbol{z} \leq \tau \quad \forall \boldsymbol{z} \in \mathrm{ext}(\mathcal{X}) \backslash \{\hat{\boldsymbol{x}}\}, \tag{8}$$

while any other restricting inequality is considered *non-separating*. Note that the term comes from the fact that when (8) is satisfied, $\boldsymbol{w}^\top \boldsymbol{x} = \tau$ can be considered as a separating hyperplane between $\hat{\boldsymbol{x}}$ and all (other) extreme points $\mathrm{ext}(\mathcal{X})$. For brevity, we refer to these inequalities simply as separating and non-separating inequalities.



(a) $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^a)$ (b) $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^b)$ (c) $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^c)$

Figure 3 **Three restricted linear relaxations, and their corresponding IFR approximations overlaid onto the true IFR. Figure (a) illustrates a pure-RLR generated with a separating inequality, while (b) and (c) illustrate mixed-RLRs generated with a non-separating inequality.**

Definition 2. *Any restricted linear relaxation $\mathcal{X}^{\mathrm{LR}}(\mathcal{W})$ can be classified into one of the following:*

- *Pure-RLR: $\mathcal{W}$ contains only separating inequalities.*
- *Mixed-RLR: $\mathcal{W}$ contains at least one non-separating inequality.*

Example 2. *Figure 3 illustrates three RLRs, generated simply by decreasing the right-hand-size value of the valid inequalities in Figures 2a and 2b. Specifically, Figure 3a illustrates a pure-RLR, where the valid inequality from Figure 2a is made into a separating inequality. Figure 3b illustrates a mixed-RLR, where the depth of the same inequality is increased until it becomes a non-separating inequality that excludes the extreme point $\boldsymbol{x}^e \in \mathrm{ext}(\mathcal{X})$ shown. Finally, Figure 3c illustrates another mixed-RLR defined by shifting the valid inequality from Figure 2b, which also excludes $\boldsymbol{x}^e$.*

4.2. Foundations and main insights

In this section, we lay the foundations of RLRs by establishing their basic properties and illustrating their role in approximating the IFR. Specifically, given a restricted linear relaxation $\mathcal{X}^{\mathrm{LR}}(\mathcal{W})$, we consider the approximation of $\mathcal{R}_{\mathrm{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ given by $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W}))$. Note that since $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}) \subseteq \mathcal{X}^{\mathrm{LR}}$, it follows from (2) that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}) \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W}))$. Furthermore, since $\mathcal{X}^{\mathrm{LR}}(\mathcal{W})$ is a polyhedral set, $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W}))$ also admits a tractable polyhedral reformulation (much like Observation 4).

Proposition 1. *Let $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}) := \{\boldsymbol{x} \in \mathbb{R}^n \mid \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}, \mathbf{W}\boldsymbol{x} \leq \boldsymbol{\tau}\}$ denote a restricted linear relaxation. Then, $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W}))$ can be reformulated as*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W})) = \left\{ \boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{p}^\top \boldsymbol{b} + \boldsymbol{z}^\top \boldsymbol{\tau},\ \mathbf{A}^\top \boldsymbol{p} + \mathbf{W}^\top \boldsymbol{z} = \boldsymbol{\theta},\ \boldsymbol{p} \geq \mathbf{0},\ \boldsymbol{z} \geq \mathbf{0} \right\}. \tag{9}$$

In Figure 3, the approximation $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W}))$ generated by each RLR is shown in the top right corner of each subfigure, overlaid onto the inverse-feasible region $\mathcal{R}_{\mathrm{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$. First, we note that the approximations in each subfigure captures a sizeable portion of $\mathcal{R}_{\mathrm{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, in contrast

to those of Figures 2a and 2b, which contains only a single inverse-feasible vector. Second, we observe that in Figure 3a, $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^a))$ forms a valid inner approximation of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, whereas $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^b))$ and $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^c))$ in Figures 3b and 3c include a small region of vectors that are not inverse-feasible. We now present two results that explain these two observations.

Theorem 1. *For any restricted linear relaxation $\mathcal{X}^{\text{LR}}(\mathcal{W})$, the following three conditions hold:*

- $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) = n$,
- $\boldsymbol{w} \in \text{int}(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})))$ *for all* $\boldsymbol{w} \in \mathbf{W}$,
- $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \cap \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})) = \dim(\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}))$.

Theorem 1 highlights that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ is a full dimensional cone in $\mathbb{R}^n$ encompassing the vectors $\boldsymbol{w}_i$ in its interior. Furthermore, its intersection with $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ spans the same subspace as $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, implying that RLRs always yield a substantive approximation of the inverse-feasible region. Figures 3a, 3b and 3c show that each approximation captures a sizeable portion of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, in contrast to those generated by the linear relaxations in Figures 2a and 2b.

Theorem 2. *For any restricted linear relaxation $\mathcal{X}^{\text{LR}}(\mathcal{W})$, the condition*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) \tag{10}$$

is satisfied if and only if $\mathcal{X}^{\text{LR}}(\mathcal{W})$ is a pure-RLR.

Theorem 2 specifies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ is an inner approximation of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ if and only if $\mathcal{W}$ consists of only separating inequalities. This explains why the IFR approximation in Figure 3a lies strictly within $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, whereas those of Figure 3b and 3c contain inverse-infeasible vectors.

Practically, RLRs provide a mechanism to construct IFR approximations of varying sizes, where smaller RLRs always yield approximations that are at least as large – and in most cases strictly larger – than those generated by larger RLRs (this follows directly from the definition of $\mathcal{K}(\hat{\boldsymbol{x}}, .)$, where, for any $\mathcal{X}' \subset \mathcal{X}'' \subset \mathcal{X}^{\text{LR}}$, it must be true that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}'') \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}')$). For example, when the depth of the restricting inequality is increased between Figures 3a and 3b, the size of the corresponding approximation $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ becomes larger, encompassing more of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ but also including vectors lying outside of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$.

Up to this point, we have relied on simple illustrative examples to develop intuition for RLRs. We now present a richer example of an SPA problem to better highlight the nuances associated with changes in the depth of a restricting inequality and, more generally, the size of an RLR.

Example 3. *Consider the classic capital budgeting problem, which aims to determine an optimal set of project investments that maximizes the total revenue from the selected projects (Petersen 1967). This problem is formulated as*

$$\begin{aligned} \max_{\boldsymbol{x}} \quad & \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \\ s.t. \quad & \boldsymbol{x} \in \mathcal{X} := \left\{ \boldsymbol{x} \in \{0,1\}^n \,\middle|\, \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b} \right\}, \end{aligned} \tag{11}$$

where $\hat{\boldsymbol{\theta}}$ denotes the predicted revenue of the projects, $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}$ represents the budget constraints, and $\boldsymbol{x} \in \mathcal{X}$ indicates the binary investment decisions. Let $\hat{\boldsymbol{x}} \in \mathcal{X}$ denote an optimal solution to this problem. For this solution, recall that we can generate exact sensitivity ranges $[\hat{\theta}_i + \Delta_i^-, \hat{\theta}_i + \Delta_i^+]$ for each coefficient $\hat{\theta}_i$, holding all others fixed, by solving a pair of inverse optimization problems (models (1a) and (1b)). Using these exact ranges as a benchmark, we examine the ranges that are generated when we replace $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ with approximations $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ generated by different RLRs. We note that for binary programs, the sensitivity range for each coefficient is always unbounded in one direction, i.e., $\Delta_i^- = -\infty$ when $\hat{x}_i = 0$ and $\Delta_i^+ = \infty$ when $\hat{x}_i = 1$, and we thus consider only the range in the bounded direction (e.g., $[\hat{\theta}_i, \hat{\theta}_i + \Delta_i^+]$ when $\hat{x}_i = 0$).

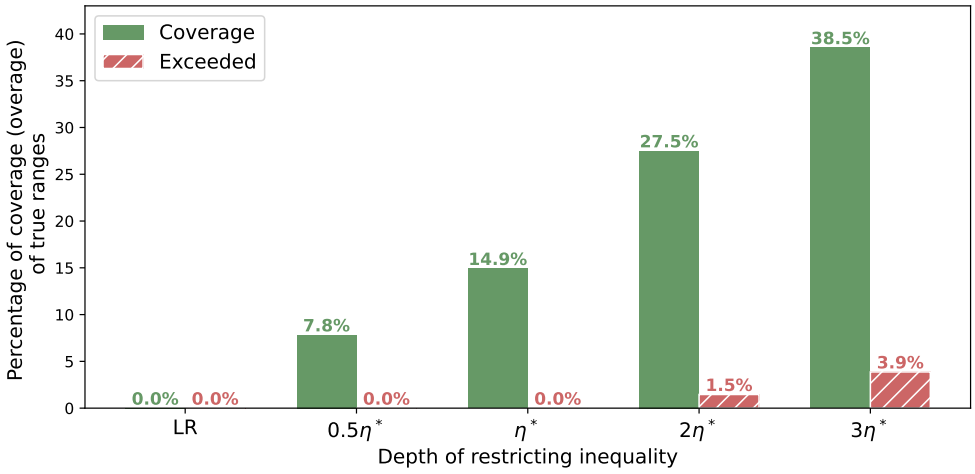


Figure 4 **The percentage of total sensitivity ranges that are covered, and the amount that exceeds the range.**

Figure 4 provides a summary of the sensitivity ranges generated for a well-known capital budgeting problem from Petersen (1967), containing 50 variables and 5 budget constraints. Specifically, we consider ranges generated by $\mathcal{X}^{\text{LR}}$, denoting the linear relaxation of (11), and by $\mathcal{X}^{\text{LR}}(\mathcal{W}) := \{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} - \eta\}$ for four different values of the depth parameter $\eta > 0$. In the figure, η^* denotes the maximal depth for which $\hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} - \eta^*$ remains a separating inequality; methods to compute η^* are deferred to Section 4.3. Each green bar shows the percentage of the true sensitivity range captured by the approximation (referred to as "coverage"), aggregated over all 50 objective coefficients. Similarly, each red striped bar shows the percentage by which the approximations exceeded the true sensitivity ranges (referred to as "overapproximation"). The complete details of this numerical example can be found in Section A2.2 of the Electronic Companion.

There are several insights that we can derive from Example 3. First, the linear relaxation $\mathcal{X}^{\text{LR}}$ fails to provide any informative approximation of the sensitivity ranges; in fact, as we illustrate

in Section A2.2, this remains true even for a tighter (but valid) linear relaxation. Second, the restricting inequality with a depth of $0.5\eta^*$ and η^* yield sensitivity ranges that remain entirely within the true ranges, with a coverage of 7.8% and 14.9%, respectively. This demonstrates both the idea that separating inequalities generate inner approximation of the IFR (Theorem 2), and that there is value in computing deeper separating inequalities when possible. Third, the results for the depths of $2\eta^*$ and $3\eta^*$ illustrate the trade-off between the level of coverage and overapproximation: increasing the depth results in approximations that cover more of the IFR while admitting a small region of inverse-infeasible vectors. The same trend is observed when moving from $2\eta^*$ and $3\eta^*$.

The examples above illustrate why RLRs outperform standard linear relaxations. More importantly, they highlight the versatility of RLRs and the practical trade-offs that arise when considering RLRs of different sizes, which will be a central theme in the remainder of this paper:

- RLRs with shallower restricting inequalities are more likely to yield inner approximations of the inverse-feasible region, though such approximations may be overly conservative.
- RLRs with deeper restricting inequalities can capture larger portions of the inverse-feasible region, but may also include regions that are not inverse-feasible.

The next two subsections establish a more complete theoretical foundation for these observations. Specifically, Section 4.3 examines the computation of the maximal depth parameter η^* beyond which a valid or separating inequality becomes non-separating, while Section 4.4 formalizes conditions under which RLRs yield strictly better approximations of the IFR. The trade-off highlighted above is also central to understanding the characteristics of the solution framework presented in Section 5, and will be further explored in the numerical experiments of Section 6.

4.3. Separating and non-separating inequalities

In this section, we more carefully consider the computation of separating and non-separating inequalities, or more specifically, the depth threshold at which a valid or separating inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$ becomes a non-separating one. For a given vector $\boldsymbol{w} \in \mathbb{R}^n$, this task can, in principle, be reduced to solving the modified forward optimization problem

$$\tau^*(\boldsymbol{w}) = \max_{\boldsymbol{x}} \ \left\{ \boldsymbol{w}^\top \boldsymbol{x} \,\middle|\, \boldsymbol{x} \in \text{ext}(\mathcal{X}) \backslash \{\hat{\boldsymbol{x}}\} \right\}. \tag{12}$$

By definition, $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$ for any $\tau < \tau^*(\boldsymbol{w})$ would define a non-separating inequality. Similarly, if $\tau^*(\boldsymbol{w}) < \boldsymbol{w}^\top \hat{\boldsymbol{x}}$, then $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$ for any $\tau \in [\tau^*(\boldsymbol{w}), \boldsymbol{w}^\top \hat{\boldsymbol{x}})$ would define a separating inequality; in this case, we refer to $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau^*(\boldsymbol{w})$ as a *tight* separating inequality. Finally, if $\tau^*(\boldsymbol{w}) = \boldsymbol{w}^\top \hat{\boldsymbol{x}}$, this implies that no separating inequality can be generated.

The main challenge in solving Problem (12) is that the set of extreme points $\text{ext}(\mathcal{X})$ is not known a priori. If it were, $\mathcal{K}(\hat{\boldsymbol{x}}, \text{ext}(\mathcal{X}))$ would provide an exact description of the IFR, and there would

be no need to solve this problem or generate approximations in the first place. As a proxy, we consider a formulation for generating separating inequalities that exclude only $\hat{\boldsymbol{x}}$ from $\mathcal{X}$, i.e.,

$$\max_{\boldsymbol{x}} \ \left\{ \boldsymbol{w}^\top \boldsymbol{x} \,\middle|\, \boldsymbol{x} \in \mathcal{X} \backslash \{\hat{\boldsymbol{x}}\} \right\}. \tag{13}$$

We note that when $\mathcal{X} \subseteq \{0,1\}^n$ is defined only by binary variables, Problems (12) and (13) are in fact equivalent, since $\mathcal{X} = \text{ext}(\mathcal{X})$. In this setting, we derive a simple mathematical program for solving them, shown in the next proposition.

Proposition 2. *When $\mathcal{X} \subseteq \{0,1\}^n$, Problem (12) and (13) can be formulated and solved as*

$$\tau^*(\boldsymbol{w}) = \max_{\boldsymbol{x}} \quad \boldsymbol{w}^\top \boldsymbol{x} \tag{14a}$$
$$\text{s.t.} \quad \boldsymbol{x} \in \mathcal{X} \tag{14b}$$
$$\sum_{i \in J_1(\hat{x}_i)} x_i + \sum_{i \in J_0(\hat{x}_i)} (1 - x_i) \leq n - 1, \tag{14c}$$

where $J_1(\hat{x}_i)$ and $J_0(\hat{x}_i)$ denote the indices $i \in \{1, \ldots, n\}$ for which $\hat{x}_i$ is equal to 1 and 0, respectively.

When $\mathcal{X} \subseteq \mathbb{Z}^n$ contains general integer variables, we can also derive a similar mathematical program to solve Problem (13), albeit one that is larger in size and requires introducing additional auxiliary variables. For brevity, we provide this formulation in **Proposition A1** in the Electronic Companion. We note that in this more general integer case where $\mathcal{X} \neq \text{ext}(\mathcal{X})$, Problem (13) is not equivalent to Problem (12), which implies that Problem (13) can still be used to generate separating inequalities but they may not be tight.

Finally, we note that separating inequalities – and, by extension, pure-RLRs – can be defined only in certain contexts; specifically, they exist only when $\hat{\boldsymbol{x}} \in \text{ext}(\mathcal{X})$. This follows directly from the observation that if $\hat{\boldsymbol{x}} \notin \text{ext}(\mathcal{X})$, then it would not be possible to separate $\hat{\boldsymbol{x}}$ from $\text{ext}(\mathcal{X})$.

4.4. Conditions for improved IFR approximations

In this final subsection, we derive mathematical conditions that further clarify when an RLR leads to a better IFR approximation compared to either $\mathcal{X}^{\text{LR}}$ or another RLR. Our analysis focuses on pure-RLRs. This is both to simplify the exposition, and because mixed-RLRs can largely be interpreted as extensions of pure-RLRs where the depth of separating inequalities are further increased; hence, the insights established for pure-RLRs remain relevant for mixed-RLRs when interpreted through this lens. Finally, all results are presented in terms of the expansion of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. Since pure-RLRs lead to inner approximations of the inverse-feasible region (Theorem 2), any expansion of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ is equivalent to capturing a larger portion of this region.

4.4.1. Improvements on $\mathcal{X}^{\text{LR}}$. We first present conditions under which an approximation $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ is better than the approximation $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$. Without loss of generality, we focus on RLRs defined by a single separating inequality.

Theorem 3. *Let $\mathcal{X}^{\text{LR}}(\mathcal{W}) := \{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \leq \tau\}$ denote an RLR defined using a separating inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$. Then,*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \tag{15}$$

if and only if there exists a $\boldsymbol{x} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{w}^\top \boldsymbol{x} > \tau$.

Put simply, Theorem 3 states that if the inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$ is able to cut off any extreme point of $\mathcal{X}^{\text{LR}}$ excluding $\hat{\boldsymbol{x}}$, then $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ will yield a strictly larger inner approximation of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ than $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$. The intuition is that extreme points of $\mathcal{X}^{\text{LR}}$ define the corresponding approximation, i.e., $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) = \mathcal{K}(\hat{\boldsymbol{x}}, \text{ext}(\mathcal{X}^{\text{LR}}))$, and eliminating at least one with a separating inequality will enlarge the approximation but preserve its validity as an inner approximation.

Proposition 3. *Let $\boldsymbol{w}$ be a vector on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$. If $\boldsymbol{w}$ can generate a separating inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$, where $\mathcal{X}^{\text{LR}}(\mathcal{W}) := \{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \leq \tau\}$, then*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})).$$

Proposition 3 provides an explanation for one of the observations made in Example 3. In this example, the nominal objective vector $\hat{\boldsymbol{\theta}}$ in fact defines a facet of (a tighter version of) $\mathcal{X}^{\text{LR}}$ and thus lies on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ (based on Observation 4). In the example, we saw that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ did not provide any informative sensitivity ranges, but the resulting $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ did. More broadly, this observation illustrates that the facets of $\mathcal{X}^{\text{LR}}$ that are tangent to $\hat{\boldsymbol{x}}$ – the normal vectors of which lie on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ – serve as good candidates for constructing restricting inequalities. Additional discussion on this example can be found in Section A2.2.

4.4.2. Improvements across RLRs. We now derive conditions for when an IFR approximation improves between two RLRs. We first focus on the depth of a single separating inequality. We show that if the addition of a separating inequality to $\mathcal{X}^{\text{LR}}$ improves the corresponding IFR approximation, then increasing its depth will result in an even better approximation.

Theorem 4. *Let $\mathcal{X}^{\text{LR}}(\mathcal{W}_1) := \{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \leq \tau_1\}$ denote an RLR where $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau_1$ is a separating inequality that is not tight (see Section 4.3). If $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$, then, for any $\mathcal{X}^{\text{LR}}(\mathcal{W}_2) := \{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \leq \tau_2\}$ where $\tau_2 < \tau_1$,*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1)) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_2)).$$

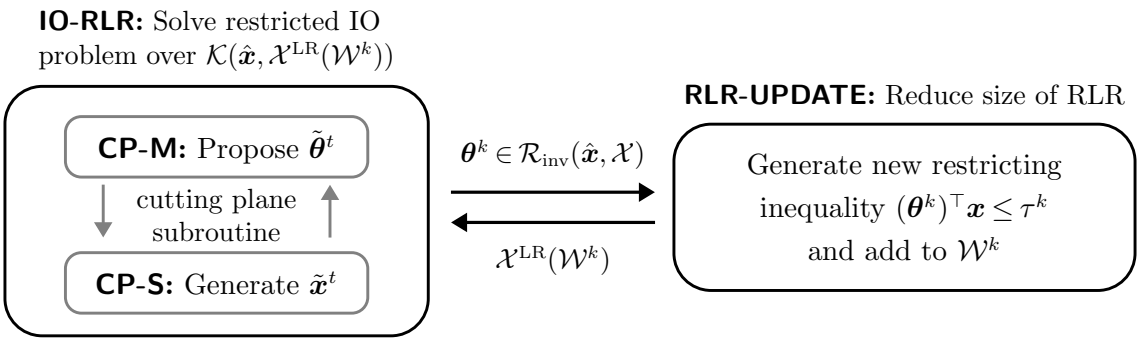


Figure 5 Visualization of the RLR-driven decomposition framework for inverse optimization.

While the discussion up to this point has focused on RLRs defined by a single separating inequality, we can extend the analysis to the case of multiple separating inequalities. Specifically, for a given $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^1)$, consider a refined $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^2) := \{\boldsymbol{x} \in \mathcal{X}^{\mathrm{LR}}(\mathcal{W}^1) \,|\, \boldsymbol{w}^\top \boldsymbol{x} \leq \tau\}$ that has an additional separating inequality. Theorem 3 naturally extends to this setting with only minor modifications.

Theorem 5. *Let* $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^2) := \{\boldsymbol{x} \in \mathcal{X}^{\mathrm{LR}}(\mathcal{W}^1) \,|\, \tilde{\boldsymbol{w}}^\top \boldsymbol{x} \leq \tilde{\tau}\}$ *denote an RLR defined by adding a separating inequality* $\tilde{\boldsymbol{w}}^\top \boldsymbol{x} \leq \tilde{\tau}$ *to an existing* $\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^1)$*. Then,*

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W}^1)) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W}^2)) \tag{16}$$

if and only if there exists a $\boldsymbol{x} \in \mathrm{ext}(\mathrm{conv}(\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^1) \cup \{\hat{\boldsymbol{x}}\}))$ *where* $\tilde{\boldsymbol{w}}^\top \boldsymbol{x} > \tilde{\tau}$*.*

The basic intuition behind this result is that $\mathrm{conv}(\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^1) \cup \{\hat{\boldsymbol{x}}\}))$ is itself a valid linear relaxation for $\mathcal{X}$, in which case Theorem 3 can be applied directly to it.

4.4.3. Summary and implications. The results above formalize several conditions under which adding restricting inequalities leads to improved IFR approximations. Taken together, they convey a simple but important message: larger portions of the inverse-feasible region can be captured either by increasing the depth of restricting inequalities or by adding more of them. This intuition serves as the conceptual basis for the IO solution methods developed in the next section.

5. RLR-Driven Algorithms for Inverse Optimization

In this section, we leverage RLRs to introduce a new framework for solving IO problems. Specifically, we propose a decomposition approach that integrates RLRs with a cutting plane subroutine, resulting in a versatile algorithmic framework that overcomes the main limitations of standalone cutting plane methods.

5.1. Description of the algorithm

We begin with a high-level summary of the approach, which we refer to as the RLR-CP Algorithm. This algorithm iteratively leverages RLRs to restrict the solution space of the inverse optimization problem. Specifically, within each iteration, a cutting plane subroutine refines the space by

eliminating regions that are not inverse-feasible, while across iterations, the space is progressively expanded by adding restricting inequalities to the linear relaxation $\mathcal{X}^{\text{LR}}$. The degree at which the solution space is expanded is determined by the depth of the restricting inequalities.

The complete details of the RLR-CP Algorithm is presented in Algorithm 1, and a schematic overview is provided in Figure 5. At its core, the RLR-CP Algorithm follows a nested decomposition structure. In each iteration of the outer loop, indexed by k, the algorithm first solves IO-RLR, which defines a restricted version of Problem (IO), i.e.,

$$\begin{aligned}
\min_{\boldsymbol{\theta},\boldsymbol{s}} \quad & f(\boldsymbol{s}) && \text{(17a)}\\
\text{s.t.} \quad & \boldsymbol{\theta} \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k)) && \text{(17b)}\\
& \boldsymbol{\theta} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) && \text{(17c)}\\
& \boldsymbol{\theta} \in \boldsymbol{\Theta}(\boldsymbol{s}) && \text{(17d)}\\
& \boldsymbol{s} \in \mathcal{S}. && \text{(17e)}
\end{aligned}$$

Solving this problem generates an inverse-feasible vector $\boldsymbol{\theta}^k \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k))$. If $f(\boldsymbol{s}^k)$ does not strictly improve on $f(\boldsymbol{s}^{k-1})$, then the RLR-CP Algorithm terminates and returns $(\boldsymbol{\theta}^k, \boldsymbol{s}^k)$ as a solution to (IO). Otherwise, the subproblem RLR-UPDATE uses $\boldsymbol{\theta}^k$ to generate a restricting inequality $(\boldsymbol{\theta}^k)^\top \boldsymbol{x} \leq \tau^k$, which is added to the existing set of restricting inequalities $\mathcal{W}^k$, creating $\mathcal{W}^{k+1}$. Recall that this restricting inequality can also be written as

$$(\boldsymbol{\theta}^k)^\top \boldsymbol{x} \leq (\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}} - \eta^k \tag{18}$$

where $\eta^k > 0$ denotes the depth of this inequality; η^k will be the focus of Sections 5.2 and 5.3.

A cutting plane subroutine is used to solve (17). This subroutine iterates between its own master problem CP-M and subproblem CP-S. At each iteration of this inner loop, indexed by t, CP-M generates a candidate vector $\tilde{\boldsymbol{\theta}}^t$ by solving a relaxed version of (17) where $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is replaced with $\mathcal{K}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}})$ for some subset of enumerated forward-feasible solutions $\tilde{\mathcal{X}} \subset \mathcal{X}$. The subproblem CP-S then tests if this candidate vector $\tilde{\boldsymbol{\theta}}^t$ is inverse-feasible. If it is not, this implies the existence of a solution $\tilde{\boldsymbol{x}}^t \in \mathcal{X}$ that is strictly better than $\hat{\boldsymbol{x}}$ under $\tilde{\boldsymbol{\theta}}^t$. This solution $\tilde{\boldsymbol{x}}^t$ is then added to $\tilde{\mathcal{X}}$ and serves as a new constraint for the next iteration of CP-M. We refer to Bodur et al. (2022) for a detailed overview of different subproblems CP-S that can be used.

5.2. Analysis and notable features

The RLR-CP Algorithm offers a versatile framework for inverse optimization. It can be viewed both as a generalization of the standard cutting plane approach, and as a method capable of operating independently from it. Specifically, as its name indicates, the algorithm integrates two components, each of which, in the extreme case, could function without the other. At one extreme, if we were to initialize $\mathcal{W}^0 = \emptyset$ as an empty set, then $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^0)) = \mathbb{R}^n$ would not restrict the

Algorithm 1: RLR-CP Algorithm for Inverse Optimization

Initialization: $k=0$, $t=0$, OptGap $=\infty$, $\Delta^0=\infty$, $u^{-1}=\infty$, $\mathcal{W}^0=\mathbb{R}^n$, $\tilde{\mathcal{X}}=\{\emptyset\}$
while $\Delta^k>0$ or OptGap $\leq\varepsilon$ **do**
  IO-RLR: repeat
    CP-M: Solve

$$\begin{aligned}\min_{\boldsymbol{\theta},\boldsymbol{s}}\quad & f(\boldsymbol{s})\\ \text{s.t.}\quad & \boldsymbol{\theta}\in\mathcal{K}(\hat{\boldsymbol{x}},\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^k))\\ & \boldsymbol{\theta}^\top(\hat{\boldsymbol{x}}-\boldsymbol{x})\geq 0\quad\forall\boldsymbol{x}\in\tilde{\mathcal{X}}\\ & \boldsymbol{\theta}\in\boldsymbol{\Theta}(\boldsymbol{s})\\ & \boldsymbol{s}\in\mathcal{S}.\end{aligned}$$

    Let $(\tilde{\boldsymbol{\theta}}^t,\tilde{\boldsymbol{s}}^t)$ denote the optimal solution.
    CP-S: Generate $\tilde{\boldsymbol{x}}^t\in\arg\max\{(\tilde{\boldsymbol{\theta}}^t)^\top\boldsymbol{x}\,|\,\boldsymbol{x}\in\mathcal{X}\}$. If $(\tilde{\boldsymbol{\theta}}^t)^\top(\tilde{\boldsymbol{x}}^t-\hat{\boldsymbol{x}})=0$, set $\tilde{\boldsymbol{x}}^t\leftarrow\emptyset$.
    $\tilde{\mathcal{X}}\leftarrow\tilde{\mathcal{X}}\cup\{\tilde{\boldsymbol{x}}^t\}$; $t\leftarrow t+1$
  until $\tilde{\boldsymbol{x}}^t=\emptyset$;
  – *Parameter updates:* $(\boldsymbol{\theta}^k,\boldsymbol{s}^k)\leftarrow(\tilde{\boldsymbol{\theta}}^t,\tilde{\boldsymbol{s}}^t)$; $u^k\leftarrow f(\boldsymbol{s}^k)$; $\Delta^k\leftarrow u^{k-1}-u^k$;
  – *Compute dual bound* ℓ^k *and optimality gap*:
    Solve IO$(\hat{\boldsymbol{x}},\tilde{\mathcal{X}})$, let ℓ^k denote optimal objective value.
    OptGap $\leftarrow\frac{u^k-\ell^k}{\ell^k}$
  RLR-UPDATE: Generate depth parameter $\eta^k>0$ and set $\tau^k\leftarrow(\boldsymbol{\theta}^k)^\top\hat{\boldsymbol{x}}-\eta^k$
    $\mathcal{W}^{k+1}\leftarrow\{\boldsymbol{x}\in\mathcal{W}^k\,|\,(\boldsymbol{\theta}^k)^\top\boldsymbol{x}\leq\tau^k\}$;
    $k\leftarrow k+1$; $t\leftarrow 0$
end
return $(\boldsymbol{\theta}^k,\boldsymbol{s}^k)$

solution space in any way, and the RLR-CP Algorithm would simply reduce to the standard cutting plane approach (i.e., the optimal solution would be generated in the first iteration of IO-RLR). On the other hand, if we initialize $\mathcal{W}^0:=\mathbb{R}^n$ without restricting inequalities, or if $\mathcal{W}^k$ were to be defined by only separating inequalities, then the corresponding $\mathcal{K}(\hat{\boldsymbol{x}},\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^k))$ is always an inner approximation of $\mathcal{R}_{\mathrm{inv}}(\hat{\boldsymbol{x}},\mathcal{X})$, rendering the cutting plane subroutine redundant for those iterations (i.e., CP-S would not generate any cuts).

Algorithmic flexibility. Given these two extremes, the depth parameter η^k provides the flexibility to operate between them, which in turn affects both the characteristics and the performance of the RLR-CP Algorithm. When the value of η^k is large, the algorithm behaves more like the standard cutting plane algorithm, since the corresponding set $\mathcal{K}(\hat{\boldsymbol{x}},\mathcal{X}^{\mathrm{LR}}(\mathcal{W}^k))$ is relatively large.

As a result, solving IO-RLR will typically require more time, but can yield significantly better solutions when solved. Conversely, when η^k is small, the algorithm will generally spend less time within the cutting plane subroutine, which results in more frequent, albeit smaller, improvements through iterations of the outer loop. In our numerical experiments, we will show that the RLR-CP Algorithm consistently achieves better performance when operating between these two extremes.

Primal and dual bounds. By design, the RLR-CP Algorithm provides a framework that can simultaneously improve primal (i.e., inverse-feasible) solutions and dual bounds for the IO problem. Specifically, the outer iterations drive improvements in the primal solutions, while the cutting plane subroutine generates feasible solutions in $\mathcal{X}$ that can be used to strengthen the dual bound. The parameter η^k can again be viewed as a lever that balances the time spent on each. As shown in Algorithm 1, this feature of the RLR-CP Algorithm permits the definition of a pre-determined optimality gap at which the solution process can be terminated. This provides a significant advantage over the standalone cutting plane approach, which returns a primal solution only at termination. In our numerical experiments, we demonstrate that the RLR-CP Algorithm can generate near-optimal solutions in far less time than is required to compute the optimal solution.

Convergence to optimality. The previous two paragraphs highlight the effect of η^k on the performance of the RLR-CP Algorithm. We now state a simple, sufficient condition on the value of η^k that ensures that the algorithm will converge to an optimal solution of Problem (IO). As stated previously, we restrict our attention to IO problems where $f(\boldsymbol{s})$ and $\mathcal{S}$ are assumed to be convex.

Proposition 4. *Suppose Problem* (IO) *is convex and bounded. If the master problem* (17) *is feasible in* $k=0$*, and if* $\eta^k > 0 \ \forall k \geq 0$*, then Algorithm 1 returns an optimal solution to* (IO)*.*

The conditions required for Proposition 4 are minimal. Put simply, the proposition states that once the first restricting inequality can be generated, the algorithm is guaranteed to return an optimal solution provided that every subsequent restricting inequality added has non-trivial depth.

5.3. Choosing the depth parameter

As discussed above, the characteristics and performance of the RLR-CP Algorithm are largely determined by the choice of η^k at each iteration. Below, we outline two general frameworks for specifying η^k. As a general guideline, when solutions to (IO) must be generated under strict time limits, smaller time limits should generally be paired with smaller depth parameters, which tend to produce faster – albeit more incremental – improvements in inverse-feasible solutions. Conversely, larger depths can provide greater improvements at the expense of increased computation time.

Static policy. We first consider a static policy, which refers to a pre-determined and fixed rule for choosing η^k. Such policies are simple and easy to implement, making it a natural starting point for understanding the behavior of the RLR-CP Algorithm; for this reason, they will be extensively tested and examined in Section 6. The most straightforward static policy is to define η^k as fixed fraction or multiple of $|(\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}}|$. Specifically, consider

$$\eta^k = \tilde{\eta} \cdot |(\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}}| \tag{19}$$

where $\tilde{\eta} > 0$ defines some fixed (i.e., static) value. This depth selection method is immune to the arbitrary scaling of $\boldsymbol{\theta}$ values, which is important since $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is a cone. Furthermore, in problem settings where $\mathcal{X} \subset \mathbb{R}^n_+$ and $\boldsymbol{\theta}^k \in \mathbb{R}^n_+$ is restricted to lie in the nonnegative orthant, $\tilde{\eta} \in [0, 1)$ is sufficient for operating between the two extremes discussed in Section 5.2.

We can also replace $|(\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}}|$ in (19) with $|(\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}} - \tau^*(\boldsymbol{\theta}^k)|$, where $\tau^*(\boldsymbol{\theta}^k)$ is the solution to Problem (13) defining the depth of a tight separating inequality. Example 3 in Section 4.1 illustrated the coverage of the inverse-feasible region when the depth parameter is defined based on a multiple of $|(\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}} - \tau^*(\boldsymbol{\theta}^k)|$. Note that the static policy based on $\tau^*(\boldsymbol{\theta}^k)$ can only be implemented for pure binary or integer forward problems (for more details, see Section 4.3).

Adaptive policy. The framework defined by the RLR-CP Algorithm also allows us to introduce a policy that adjusts the depth parameter dynamically based on the algorithm's real-time performance. This both mitigates the algorithm's sensitivity to the initial depth values specified under the static policy, and enhances its performance through a more automated parameter selection process. Due to space considerations, we differ its presentation and discussion to Section A3.1 of the Electronic Companion. To the best of our knowledge, this approach constitutes the first adaptive solution method proposed for inverse optimization.

6. Applications and Computational Insights

In this section, we first outline how inverse optimization can be applied to the sensitivity and post-optimality analysis of modern optimization problems (Section 6.1). We then focus on one such application, where we use a set of numerical experiments to derive insights into the performance and advantages of our RLR-CP Algorithm in addressing such problems (Section 6.2).

6.1. Post-optimality analysis for modern prescriptive problems

The sensitivity and post-optimality analysis (SPA) of prescriptive models aims to understand how changes in model parameters affect the optimality of a recommended decision. Although this has been a longstanding topic of study in optimization and prescriptive analytics, the SPA literature for discrete optimization is rather sparse, with almost all work focusing on the single-coefficient

SPA problem (see Section 2 and Observation 1). By reframing SPA through the lens of inverse optimization, we broaden the set of questions that can be posed and enable richer insights into prescriptive solutions generated by modern optimization pipelines. We present examples in multi-objective optimization (Section 6.1.1) and contextual optimization (Section 6.1.2), both of which demand the analysis of parameters that underlie multiple objective coefficients.

6.1.1. Multi-objective optimization. Most real-world decision-making problems involve balancing multiple, potentially conflicting objectives. This is especially true in high-stakes domains and problems related to public policy. In practice, the predominant approach for solving multi-objective optimization problems is to aggregate objectives through a weighted-sum scalarization.

Specifically, consider a discrete optimization problem with d number of linear objectives, denoted as $\boldsymbol{f}_1^\top \boldsymbol{x}, \ldots, \boldsymbol{f}_d^\top \boldsymbol{x}$. The weighted-sum scalarization of this problem can then be written as

$$\max_{\boldsymbol{x}} \quad \left\{ \sum_{i=1}^{d} s_i \boldsymbol{f}_i^\top \boldsymbol{x} \,\middle|\, \boldsymbol{x} \in \mathcal{X} \right\}, \tag{20}$$

where $s_i \geq 0$ denotes the weight assigned to the objective i. Without loss of generality, we can assume that $\sum_{i=1}^{d} s_i = 1$, where the value of s_i can be interpreted as the relative importance that is placed on the i-th objective.

Discrete optimization problems of the form (20) are widely used to support decision-making (e.g., Rönnqvist et al. 2017, Bertsimas et al. 2019, Chu et al. 2020, Peters et al. 2022, Christou et al. 2024, Ozel et al. 2025). Despite their widespread use, a key challenge that arises is the choice of the weights $\boldsymbol{s}$ for (20). This task is complicated by the fact that different stakeholders often have different preferences over these weights, and reaching a consensus can become an extremely time consuming process (e.g., Kik et al. 2023, Umang et al. 2023).

In this context, a natural SPA problem is to understand and explore the range of weight vectors $\boldsymbol{s}$ that would all lead to the same prescriptive solution. By definition, this requires extending the analysis of single objective coefficients to the analysis of weights that simultaneously influence multiple objective coefficients. We can formulate this generalized SPA problem as

$$\begin{aligned} \min_{\boldsymbol{\theta}, \boldsymbol{s}} \quad & f(\boldsymbol{s}) \\ \text{s.t.} \quad & \boldsymbol{\theta} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) \\ & \boldsymbol{\theta} = \sum_{i}^{d} s_i \boldsymbol{f}_i \\ & \boldsymbol{s} \in \mathcal{S} \end{aligned} \tag{GSPA}$$

under various choices of $f(\boldsymbol{s})$ and $\mathcal{S}$. Our numerical experiments in Section 6.2 will be based on one such specification of (GSPA), which we introduce as an example below.

Example 4. *Let $\hat{\boldsymbol{x}} \in \mathcal{X}$ be the optimal solution generated by (20) under a nominal set of weights $\hat{\boldsymbol{s}}$. Now, consider (GSPA) where $f(\boldsymbol{s}) = s_i$ (or $-s_i$) and $\mathcal{S} = \{\boldsymbol{s} \,|\, ||\boldsymbol{s}||_1 = 1, \boldsymbol{s} \geq \boldsymbol{0}\}$. This specification can be used to assess the flexibility in assigning a weight to a particular objective i. For instance, if s_i can be increased substantially compared to a nominal value $\hat{s}_i$, it implies that $\hat{\boldsymbol{x}}$ could still be generated under a much higher weighting of objective i, provided that the remaining weights are adjusted accordingly. Conversely, if s_i cannot vary by much, it suggest that a different prescriptive solution should be considered if one were to place more emphasis on objective i. More generally, we can use such specifications to explore the family of weights that ultimately yield the same solution.*

6.1.2. Contextual optimization. Problems in the form of (GSPA) are also relevant in contextual and data-driven optimization, especially in predict-then-optimize pipelines. In this literature, we use a set of contextual features $\{\boldsymbol{f}_1, \ldots, \boldsymbol{f}_m\} \in \mathbb{R}^n$ and a function $\hat{g}(\boldsymbol{f}_1, \ldots, \boldsymbol{f}_m) \to \hat{\boldsymbol{\theta}}$ to a generate a prediction for $\boldsymbol{\theta}$ in (FO). As an example, in routing problems aimed at minimizing total travel time, each feature $\boldsymbol{f}_i \in \mathbb{R}^d$ may encode information such as time-of-day, weather conditions, and traffic incidents. These contextual features are then used to predict the travel times along arcs of a road network, represented collectively by the parameter vector $\boldsymbol{\theta}$.

One of the most widely studied functions is the affine mapping, i.e.,

$$\hat{g}(\boldsymbol{f}_1, \ldots, \boldsymbol{f}_m) := \beta_0 + \beta_1 \boldsymbol{f}_1 + \ldots + \beta_d \boldsymbol{f}_m \to \hat{\boldsymbol{\theta}}, \tag{21}$$

where $\boldsymbol{\beta} = [\beta_0, \ldots, \beta_d]$ denotes the set of regression coefficients that are estimated from a data set (e.g., Elmachtoub and Grigas 2022).

Consider an optimal solution $\hat{\boldsymbol{x}}$ to FO($\hat{\boldsymbol{\theta}}, \mathcal{X}$), where $\hat{\boldsymbol{\theta}}$ is generated via (21). Much like the multi-objective problems described in Section 6.1.1, the traditional single-coefficient SPA problem for $\hat{\boldsymbol{\theta}}$ may be of limited value, since the objective coefficients are not independent but are instead coupled through the features and regression parameters. Using variations of (GSPA), e.g., where $\boldsymbol{s}$ is replaced with $\boldsymbol{\beta}$, we can examine the sensitivity of a prescriptive decision to the values of $\boldsymbol{\beta}$. Similarly, if we hold $\boldsymbol{\beta}$ fixed but define $\boldsymbol{f}$ to be decision variables in the inverse problem, we could analyze the sensitivity to the contextual features $\boldsymbol{f}$.

6.2. Computational study

We present a computational study to explore the characteristics and performance of the RLR-CP Algorithm when solving SPA problems defined in the form of (GSPA). We consider forward problems that are multi-objective knapsack problems, which commonly arise in resource allocation and selection problems (e.g., Montoya-Weiss and Calantone 1999, Kulturel-Konak et al. 2023). We first outline the experimental setup, followed by the main discussion and insights.

6.2.1. Experimental setup. We consider multi-objective knapsack problems in the form of (20) with 100, 200, 400, and 800 variables, each with 10 objective vectors. For each problem, we generate a solution $\hat{\boldsymbol{x}}$ using a nominal uniform weighting of $\hat{s}_i = 0.1$ for each objective $i \in \{1, \ldots, 10\}$. Then, we compute the maximum values that each s_i could take while $\hat{\boldsymbol{x}}$ remains optimal, by solving the problem described in Example 4. This leads to a total of 40 inverse discrete optimization problems spanning the four problem sizes.

For the RLR–CP Algorithm, we consider a static depth selection policy based on (19), where $\tilde{\eta} = \{0.0001, 0.001, 0.01, 0.1\}$. We adopt the trust-region based method from Bodur et al. (2022) as the CP-subroutine in our RLR–CP algorithm. Similarly, as benchmarks, we consider the standard CP algorithm (Wang 2009) and the trust-region based CP algorithm (Bodur et al. 2022). These simply correspond to versions of the RLR-CP Algorithm without the RLR component, i.e., where $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\mathrm{LR}}(\mathcal{W})) = \mathbb{R}^n$ (see Section 5.2).

The full details of the instances, the specifications of the RLR-CP Algorithm, and the benchmarks are found in Section A4.2 of the Electronic Companion. All experiments are conducted on an Apple Mac M4 equipped with 10 cores and 10 threads and 16GBs of RAM using Gurobi 12.0.1 in Python.

6.2.2. Analysis and insights. We begin by summarizing the three key insights that are demonstrated through our computational experiments. The corresponding results and discussion that substantiate and elaborate on each insight is presented after the summary.

- *Insight I:* Under tight time constraints, the RLR-CP Algorithm can generate informative and high-quality solutions to (IO), whereas standalone CP algorithms are of little practical value.
- *Insight II:* As a general guideline, shorter time constraints should be paired with smaller depth parameters, as they tend to yield more frequent improvements in solution quality.
- *Insight III:* There exists a critical point in which further decreasing the depth parameter may degrade the performance of the RLR-CP Algorithm. This also suggests that the algorithm could further benefit from a more systematic tuning of the depth selection policy.

Tables 2 and 3 illustrates the aggregate results for solving the SPA problems over the instances with 200 and 400 variables, respectively; results for the other instances can be found in Section A4.3 in the Electronic Companion. Each table highlights the number of instances, out of 10, that are solved to a specified optimality gap within a given time limit.

We first use Tables 2 and 3 to examine the performance the RLR-CP Algorithm relative to standalone CP algorithms. Recall that CP algorithms do not generate (or improve) inverse-feasible solutions until the final iteration, at which point it generates the optimal one. Because of this property, CP algorithms can be entirely uninformative under time constraints. For example, when we consider a time limit of 25 seconds in Table 2 or 500 seconds in Table 3, CP algorithms do

Time Limit	**25 seconds**				**50 seconds**				**100 seconds**			
Optimality Gap	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**
RLR–CP: $\tilde{\eta} = 0.0001$	3	0	0	0	9	5	1	0	10	10	9	4
RLR–CP: $\tilde{\eta} = 0.001$	9	7	1	1	9	9	9	9	10	10	10	10
RLR–CP: $\tilde{\eta} = 0.01$	0	0	0	0	3	3	3	3	9	9	9	9
RLR–CP: $\tilde{\eta} = 0.1$	0	0	0	0	0	0	0	0	5	5	5	5
CP: Standard	0	0	0	0	0	0	0	0	6	6	6	6
CP: Trust Region	0	0	0	0	3	3	3	3	9	9	9	9

Table 2 **Number of instances reaching specified optimality gaps within time limits, for 200 variable instances.**

Time Limit	**200 seconds**				**500 seconds**				**1000 seconds**			
Optimality Gap	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**
RLR–CP: $\tilde{\eta} = 0.0001$	10	9	3	0	10	10	10	3	10	10	10	9
RLR–CP: $\tilde{\eta} = 0.001$	3	2	1	0	9	8	7	7	10	10	10	10
RLR–CP: $\tilde{\eta} = 0.01$	0	0	0	0	1	1	1	1	8	8	8	8
RLR–CP: $\tilde{\eta} = 0.1$	0	0	0	0	0	0	0	0	5	5	5	5
CP: Standard	0	0	0	0	0	0	0	0	6	6	6	6
CP: Trust Region	0	0	0	0	0	0	0	0	7	7	7	7

Table 3 **Number of instances reaching specified optimality gaps within time limits, for 400 variable instances.**

not generate any inverse-feasible solution within any level of optimality. In contrast, the RLR-CP Algorithm with smaller values of $\tilde{\eta}$ can generate near-optimal solutions within the same allotted time. For example, in 500 seconds, all instances in Table 3 could be solved to 1% optimality for the smallest value of $\tilde{\eta}$. Furthermore, the RLR-CP Algorithm can progressively refine these inverse-feasible solutions over time; by 1000 seconds, nearly all instances across all values of $\tilde{\eta}$ are solved to optimality. These results demonstrate that the RLR-CP Algorithm is effective both at efficiently generating high-quality solutions and at solving inverse optimization problems to optimality.

The second insight that can be drawn from these tables is that increasing the depth parameter can potentially lead to a significant degradation in performance. This pattern is consistent with both the theoretical results and our expectations: as the value of $\tilde{\eta}$ increases, the RLR-CP Algorithm behaves more like the standalone CP algorithm, which suffers from the limitations discussed in the previous paragraph (and also in Section 5.2). This is reflected in the results of $\tilde{\eta} = 0.1$. Conversely, when we decrease the depth parameter too much beyond a certain point, we also diminish the ability to make significant improvements. This is reflected in the results of $\tilde{\eta} = 0.0001$ in Table 2.

Figure 6 provides a more detailed illustration on the performance of each algorithmic specification for a particular instance of the 200-variable model. In this instance, we seek to maximize the value of the weight of the 7-th objective vector, s_7, in (GSPA). In the optimal solution $\boldsymbol{s}^*$, $s_7^* = 0.1764$.

The figure shows a plot of the values of $\tilde{s}_7^t$ in all candidate vectors $\tilde{\boldsymbol{s}}^t$ that are generated in each iteration of the cutting plane subroutine, as well as those of the standalone CP algorithm.

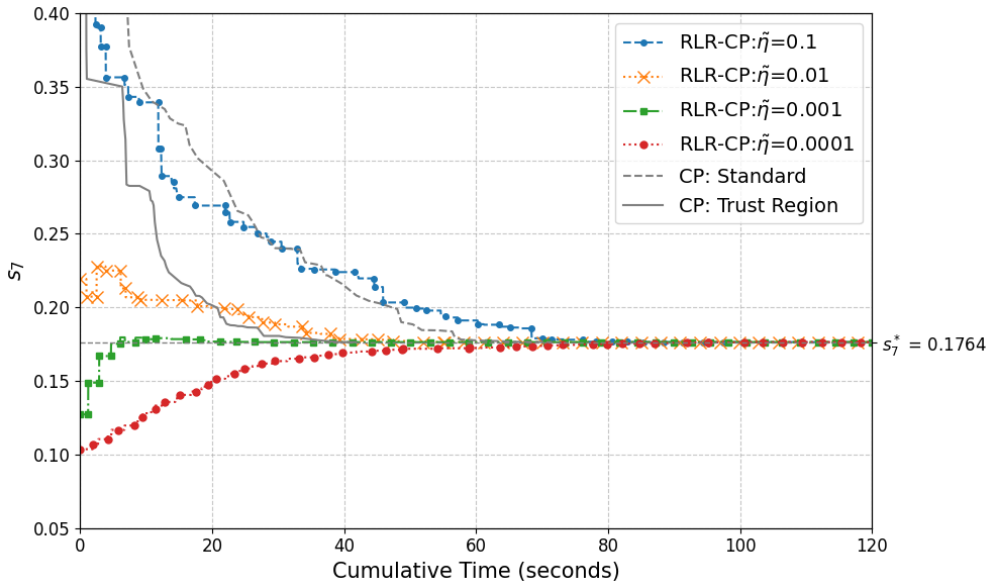


Figure 6 **The performance of different algorithms for a single 200 variable instance.**

To best interpret this figure, we first note that all values above s_7^* correspond to candidate vectors $\boldsymbol{s}^t$ that are not inverse-feasible (otherwise, s_7^* cannot be optimal). On the other hand, for the line segments that lie below s_7^*, any point on the line that precedes a shift towards s_7^* must correspond to an inverse-feasible vector (as the increase can only result from adding a new restricting inequality using a new inverse-feasible vector). With this in mind, we first remark that both CP algorithms, as expected, converge towards the value of s_7^* from above. This is also true for the RLR-CP Algorithm for larger values of $\tilde{\eta}$, i.e., when $\tilde{\eta} = 0.1$ and $\tilde{\eta} = 0.01$. However, when we further decrease $\tilde{\eta}$ to 0.001, the algorithm initially generates a few inverse-feasible vectors, followed by a few cuts, and then quickly converges to s_7^*. Finally, when we further decrease $\tilde{\eta}$ to 0.0001, the algorithm continues to generate and improve inverse-feasible vectors, albeit at a slower rate. These observations also demonstrate why it can be useful to view the RLR-CP Algorithm as a more complete primal-dual solution framework for inverse optimization, where the depth parameter implicitly controls the time spent improving dual bounds versus primal (i.e., inverse-feasible) solutions.

7. Conclusion

Inverse optimization has emerged as a powerful framework for analyzing model parameters that rationalize observed or prescribed decisions. In this paper, we take a holistic approach to solving and applying IO models for prescriptive problems that have discrete decision variables, and make

three main contributions. We first introduce a versatile and novel technique for generating rich approximations of the inverse-feasible region using the concept of restricting linear relaxations. We then leverage RLRs to develop a generalized solution framework for inverse optimization, and show that this framework overcomes the main limitation of standalone cutting plane approaches. Finally, we use inverse optimization to develop a modern perspective on sensitivity and post-optimality analysis, through which we derive insights into the models and algorithms proposed in the paper. We conclude by providing two general directions for future research.

Direct extensions. The primary goal of our paper is to propose new ideas and methods for inverse optimization that are broadly applicable across discrete optimization problems. Nevertheless, it is natural to expect that these ideas can potentially be further refined by focusing on specific classes of discrete forward problems (e.g., knapsack, matching, packing, routing). In such settings, it may be possible to derive stronger theoretical results or integrate existing techniques – such as problem-specific valid inequalities – into the methodological frameworks developed here.

Broader applications. New applications of inverse optimization and the inverse-feasible region continue to emerge. Exploring problems beyond those studied in this paper may, in turn, motivate new methodological questions that require extending the approaches proposed here. For example, Section 2.3 highlights several applications that rely on the inverse-feasible region but that do not align precisely with the IO formulations considered in this paper. These examples underscore the broader relevance of the ideas and the opportunities for extending them to new domains.

References

Agarwal R, Ergun Ö (2010) Network design and allocation mechanisms for carrier alliances in liner shipping. *Operations Research* 58(6):1726–1742.

Ahuja RK, Orlin JB (2001) Inverse optimization. *Operations Research* 49(5):771–783.

Andersen KA, Boomsma TK, Nielsen LR (2023) MILP sensitivity analysis for the objective function coefficients. *INFORMS Journal on Optimization* 5(1):92–109.

Aswani A, Shen ZJ, Siddiq A (2018) Inverse optimization with noisy data. *Operations Research* 66(3):870–892.

Berden S, Mahmutoğulları Aİ, Tsouros D, Guns T (2025) Solver-free decision-focused learning for linear optimization problems. *arXiv preprint arXiv:2505.22224* .

Bertsimas D, Delarue A, Martin S (2019) Optimizing schools' start time and bus routes. *Proceedings of the National Academy of Sciences* 116(13):5943–5948.

Bertsimas D, Gupta V, Paschalidis IC (2015) Data-driven estimation in equilibrium using inverse optimization. *Mathematical Programming* 153(2):595–633.

Bertsimas D, Tsitsiklis JN (1997) *Introduction to linear optimization*, volume 6 (Athena scientific Belmont, MA).

Besbes O, Fonseca Y, Lobel I (2025) Contextual inverse optimization: Offline and online learning. *Operations Research* 73(1):424–443.

Bodur M, Chan TC, Zhu IY (2022) Inverse mixed integer optimization: Polyhedral insights and trust region methods. *INFORMS Journal on Computing* 34(3):1471–1488.

Bulut A, Ralphs TK (2021) On the complexity of inverse mixed integer linear optimization. *SIAM Journal on Optimization* 31(4):3014–3043.

Chan T, Delage E, Lin B (2024) Conformal inverse optimization for adherence-aware prescriptive analytics. *Available at SSRN* .

Chan TC, Mahmood R, Zhu IY (2025) Inverse optimization: Theory and applications. *Operations Research* 73(2):1046–1074.

Christou IT, Vagianou E, Vardoulias G (2024) Planning courses for student success at the american college of greece. *INFORMS Journal on Applied Analytics* 54(4):365–379.

Chu A, Keskinocak P, Villarreal MC (2020) Empowering denver public schools to optimize school bus operations. *INFORMS Journal on Applied Analytics* 50(5):298–312.

Chvátal V, Cook W, Espinoza D (2013) Local cuts for mixed-integer programming. *Mathematical Programming Computation* 5(2):171–200.

Cornuéjols G (2008) Valid inequalities for mixed integer linear programs. *Mathematical Programming* 112(1):3–44.

Duan Z, Wang L (2011) Heuristic algorithms for the inverse mixed integer linear programming problem. *Journal of Global Optimization* 51:463–471.

D'Ambrosio C, Frangioni A, Liberti L, Lodi A (2010) On interval-subgradient and no-good cuts. *Operations Research Letters* 38(5):341–345.

Elmachtoub AN, Grigas P (2022) Smart "predict, then optimize". *Management Science* 68(1):9–26.

Geoffrion AM, Nauss R (1977) Parametric and postoptimality analysis in integer linear programming. *Management Science* 23(5):453–466.

Heuberger C (2004) Inverse combinatorial optimization: a survey on problems, methods, and results. *Journal of Combinatorial Optimization* 8:329–361.

Jenkins L (1982) Parametric mixed integer programming: an application to solid waste management. *Management Science* 28(11):1270–1284.

Kik D, Wichmann MG, Spengler TS (2023) Small-or medium-sized enterprise uses operations research to select and develop its headquarters location. *INFORMS Journal on Applied Analytics* 53(4):312–331.

Kılınç-Karzan F, Toriello A, Ahmed S, Nemhauser G, Savelsbergh M (2009) Approximating the stability region for binary mixed-integer programs. *Operations Research Letters* 37(4):250–254.

Klein D, Holm S (1979) Integer programming post-optimal analysis with cutting planes. *Management Science* 25(1):64–72.

Kulturel-Konak S, Konak A, Jakielaszek L, Gavirneni N (2023) Menu engineering for continuing care senior living facilities with captive dining patrons. *INFORMS Journal on Applied Analytics* 53(3):218–239.

Lamperski JB, Schaefer AJ (2015) A polyhedral characterization of the inverse-feasible region of a mixed-integer program. *Operations Research Letters* 43(6):575–578.

Lin B, Delage E, Chan T (2024) Conformal inverse optimization. *Advances in Neural Information Processing Systems* 37:63534–63564.

Liu L, Qi X, Xu Z (2024) Stabilizing grand cooperation via cost adjustment: An inverse optimization approach. *INFORMS Journal on Computing* 36(2):635–656.

Marcotte P, Mercier A, Savard G, Verter V (2009) Toll policies for mitigating hazardous materials transport risk. *Transportation Science* 43(2):228–243.

Mohajerin Esfahani P, Shafieezadeh-Abadeh S, Hanasusanto GA, Kuhn D (2018) Data-driven inverse optimization with imperfect information. *Mathematical Programming* 167(1):191–234.

Montoya-Weiss M, Calantone RJ (1999) Development and implementation of a segment selection procedure for industrial product markets. *Marketing Science* 18(3):373–395.

Niendorf M, Girard AR (2017) Exact and approximate stability of solutions to traveling salesman problems. *IEEE Transactions on Cybernetics* 48(2):583–595.

Niendorf M, Kabamba PT, Girard AR (2015) Stability of solutions to classes of traveling salesman problems. *IEEE Transactions on Cybernetics* 46(4):973–985.

Ozel A, Smilowitz K, Goldstein LK (2025) Community-engaged school district design: A stream-based approach. *Operations Research* (Articles in Advance).

Peters K, Silva S, Wolter TS, Anjos L, van Ettekoven N, Combette É, Melchiori A, Fleuren H, den Hertog D, Ergun Ö (2022) UN world food programme: toward zero hunger with analytics. *INFORMS journal on applied analytics* 52(1):8–26.

Petersen CC (1967) Computational experience with variants of the Balas algorithm applied to the selection of R&D projects. *Management Science* 13(9):736–750.

Rönnqvist M, Svenson G, Flisberg P, Jönsson LE (2017) Calibrated route finder: Improving the safety, environmental consciousness, and cost effectiveness of truck routing in sweden. *Interfaces* 47(5):372–395.

Roodman GM (1974) Postoptimality analysis in integer programming by implicit enumeration: The mixed integer case. *Naval Research Logistics Quarterly* 21(4):595–607.

Schaefer AJ (2009) Inverse integer programming. *Optimization Letters* 3:483–489.

Schrage L, Wolsey L (1985) Sensitivity analysis for branch and bound integer programming. *Operations Research* 33(5):1008–1023.

Shahmoradi Z, Lee T (2022) Quantile inverse optimization: Improving stability in inverse linear programming. *Operations research* 70(4):2538–2562.

Tang B, Khalil EB (2024) Cave: A cone-aligned approach for fast predict-then-optimize with binary linear programs. *International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, 193–210 (Springer).

Tavaslıoğlu O, Lee T, Valeva S, Schaefer AJ (2018) On the structure of the inverse-feasible region of a linear program. *Operations Research Letters* 46(1):147–152.

Umang N, Kiefer D, Gonzalez P, Thevenot H, Johnson CD, Martin J, Stephenson E, Cline J, Gemici-Ozkan B, Beniaminy I (2023) Practice summary: GE optimizes for aircraft engine overhaul scheduling and shop assignment. *INFORMS Journal on Applied Analytics* 53(2):128–132.

Wang L (2009) Cutting plane algorithms for the inverse mixed integer linear programming problem. *Operations Research Letters* 37(2):114–116.

Wei N, Zhang P (2024) Adjustability in robust linear optimization. *Mathematical Programming* 208(1):581–628.

Zattoni Scroccaro P, Atasoy B, Mohajerin Esfahani P (2025) Learning in inverse optimization: Incenter cost, augmented suboptimality loss, and algorithms. *Operations Research* 73(5):2661–2679.

Zhou W, Orfanoudaki A, Zhu S (2025) Conformalized decision risk assessment. *arXiv preprint arXiv:2505.13243* .

Electronic Companion to "Inverse Optimization with Discrete Decisions"

Section A1 contains the proofs of all results presented in the main body of the paper. Sections A2, A3 and A4 contain additional materials for Sections 4, 5 and 6, respectively.

A1. Proofs

Proofs are presented in the same order as the results in the paper.

A1.1. Proof of Proposition 1

We can reformulate the cone $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ as follows, starting with its original definition:

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) = \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x}, \quad \forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})\right\} \tag{A1a}$$

$$= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \max\{\boldsymbol{\theta}^\top \boldsymbol{x} \,|\, \forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})\}\right\} \tag{A1b}$$

$$= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \max\{\boldsymbol{\theta}^\top \boldsymbol{x} \,|\, \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}, \mathbf{W}\boldsymbol{x} \leq \boldsymbol{\tau}\}\right\} \tag{A1c}$$

$$= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \min\{\boldsymbol{p}^\top \boldsymbol{b} + \boldsymbol{z}^\top \boldsymbol{\tau} \,|\, \mathbf{A}^\top \boldsymbol{p} + \mathbf{W}^\top \boldsymbol{z} = \boldsymbol{\theta}, \boldsymbol{p} \geq \mathbf{0}, \boldsymbol{z} \geq \mathbf{0}\}\right\} \tag{A1d}$$

$$= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{p}^\top \boldsymbol{b} + \boldsymbol{z}^\top \boldsymbol{\tau},\ \mathbf{A}^\top \boldsymbol{p} + \mathbf{W}^\top \boldsymbol{z} = \boldsymbol{\theta}, \boldsymbol{p} \geq \mathbf{0}, \boldsymbol{z} \geq \mathbf{0}\right\} \tag{A1e}$$

The reformulation from (A1c) to (A1d) uses strong duality, where $\boldsymbol{p}$ and $\boldsymbol{z}$ are the dual variables associated with constraints $\mathbf{A}\boldsymbol{x} \leq \boldsymbol{b}$ and $\mathbf{W}\boldsymbol{x} \leq \boldsymbol{\tau}$, respectively.

A1.2. Proof of Theorem 1

1. Proof that $\boldsymbol{w} \in \text{int}(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ for all $\boldsymbol{w} \in \mathbf{W}$: Note that since $\mathcal{X}^{\text{LR}}(\mathcal{W})$ is defined by restricting inequalities $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$ where $\boldsymbol{w}^\top \hat{\boldsymbol{x}} > \tau$, it must be true that $\boldsymbol{w}^\top \hat{\boldsymbol{x}} > \boldsymbol{w}^\top \boldsymbol{x}$ must hold for all $\boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})$. Since there are no binding constraints at $\boldsymbol{w}$, this implies that $\boldsymbol{w} \in \text{int}(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})))$.

2. Proof that $\dim(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) = n$: Since there exists a $\boldsymbol{w} \in \text{int}(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})))$, this also implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ must be full-dimensional, i.e., $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})) = n$.

3. Proof that $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \cap \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) = \dim(\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}))$: Suppose $\dim(\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})) = m$ where $m \leq n$. This implies that we can generate m number of linearly independent vectors $\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^m \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$. Consider any $\boldsymbol{w} \in \mathbf{W}$. Since $\boldsymbol{w} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ and $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is convex, any convex combination of $\boldsymbol{w}$ and $\boldsymbol{\theta}^i$ where $i = \{1, \ldots, m\}$ must also be inverse-feasible. Furthermore, note that since $\boldsymbol{w} \in \text{int}(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})))$, there must exist a $\mathcal{B}_\epsilon(\boldsymbol{w}) \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$ where $\mathcal{B}_\epsilon(\boldsymbol{w}) := \{\boldsymbol{w} + \boldsymbol{\delta} \,|\, ||\boldsymbol{\delta}||_2 \leq \varepsilon\}$ for some $\epsilon > 0$. Thus, for any vector $(\boldsymbol{\theta}^i - \boldsymbol{w})$, there must exist a sufficiently small $0 < \lambda^i \leq 1$ for which $\boldsymbol{w} + \lambda^i(\boldsymbol{\theta}^i - \boldsymbol{w})$ lies in $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$; note that this $\boldsymbol{w} + \lambda^i(\boldsymbol{\theta}^i - \boldsymbol{w})$ also belongs to $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, as it is a convex combination of $\boldsymbol{w}$ and $\boldsymbol{\theta}^i$. Finally, since

$$\text{span}\big(\boldsymbol{w}, \boldsymbol{w} + \lambda^1(\boldsymbol{\theta}^1 - \boldsymbol{w}), \ldots, \boldsymbol{w} + \lambda^m(\boldsymbol{\theta}^m - \boldsymbol{w})\big) = m, \tag{A2}$$

it must be true that $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \cap \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})) = m$. □

A1.3. Proof of Theorem 2

($\Leftarrow$) We first prove that if $\mathcal{X}^{\text{LR}}(\mathcal{W})$ is a pure-RLR, then $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$. The main idea of this proof is to leverage the fact that $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is equivalent to $\mathcal{K}(\hat{\boldsymbol{x}}, \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\})$, i.e.,

$$\begin{aligned}
\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) &= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x} \quad \forall \boldsymbol{x} \in \mathcal{X}\right\} \\
&= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x} \quad \forall \boldsymbol{x} \in \text{conv}(\mathcal{X})\right\} \\
&= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x} \quad \forall \boldsymbol{x} \in \text{ext}(\mathcal{X})\right\} \\
&= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x} \quad \forall \boldsymbol{x} \in \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}\right\} \\
&= \mathcal{K}(\hat{\boldsymbol{x}}, \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}).
\end{aligned}$$

Let $\tilde{\mathcal{X}} := \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}$. Since $\mathcal{X}^{\text{LR}}(\mathcal{W})$ is a pure-RLR, $\tilde{\mathcal{X}} \subset \mathcal{X}^{\text{LR}}(\mathcal{W})$ by definition. Thus, it must be true that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}}) = \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, as $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}') \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}'')$ for any $\mathcal{X}'' \subseteq \mathcal{X}'$.

($\Rightarrow$) We now show that if $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ then $\mathcal{X}^{\text{LR}}(\mathcal{W})$ must be a pure-RLR. We consider a proof by contradiction, namely, that there exists a mixed-RLR $\mathcal{X}^{\text{LR}}(\mathcal{W})$ for which $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$.

In order to prove this statement, we first note that, by Theorem 1, since $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))) = n$, it must be true that $\dim(\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})) = n$ and that $\hat{\boldsymbol{x}} \in \text{ext}(\mathcal{X})$. Furthermore, since $\mathcal{X}^{\text{LR}}(\mathcal{W})$ is a mixed-RLR, it must be true that there exists a $\tilde{\boldsymbol{x}} \in \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}$ such that $\tilde{\boldsymbol{x}} \notin \mathcal{X}^{\text{LR}}(\mathcal{W})$.

The proof follows in three parts:

Part 1 We first show that if there exists a $\tilde{\boldsymbol{x}} \in \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{w}^\top \tilde{\boldsymbol{x}} > \tau$, then there must also exist a $\bar{\boldsymbol{x}} \in \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}$ that is adjacent to $\hat{\boldsymbol{x}}$ that also satisfies $\boldsymbol{w}^\top \bar{\boldsymbol{x}} > \tau$.

Part 2 We then show that, for this $\bar{\boldsymbol{x}} \in \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}$, there exists a $\tilde{\boldsymbol{\theta}}$ such that $\bar{\boldsymbol{x}}$ is the only solution to $\max\{\tilde{\boldsymbol{\theta}}^\top \boldsymbol{x} \,|\, \boldsymbol{x} \in \text{ext}(\mathcal{X}) \setminus \{\hat{\boldsymbol{x}}\}\}$ where $\tilde{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} = \tilde{\boldsymbol{\theta}}^\top \bar{\boldsymbol{x}}$. We show that this $\tilde{\boldsymbol{\theta}}$, by definition, must be on the boundary of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$.

Part 3 Finally, we show that this particular $\tilde{\boldsymbol{\theta}}$ is in the interior of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. This implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \not\subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$.

Proofs of Parts 1 and 2: To avoid repetition, the proofs of Parts 1 and 2 follow directly from the analogous arguments in Parts 1 and 2 of Section A1.5, by replacing $\mathcal{X}^{\text{LR}}$ in A1.5 with $\text{conv}(\mathcal{X})$, since $\text{ext}(\mathcal{X})$ is equivalent to $\text{ext}(\text{conv}(\mathcal{X}))$ (Section 1.2).

Proof of Part 3: We first show that there cannot exist any $\boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})$ for which $\tilde{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} = \tilde{\boldsymbol{\theta}}^\top \boldsymbol{x}$. To do so, we define the line segment between $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$ as $\mathcal{L} := \{\boldsymbol{y} \,|\, \boldsymbol{y} = \rho\hat{\boldsymbol{x}} + (1-\rho)(\bar{\boldsymbol{x}}),\ \rho \in [0,1]\}$. Note that by the definition of $\tilde{\boldsymbol{\theta}}$, $\mathcal{L} \subseteq \text{conv}(\mathcal{X})$ describes the complete set of solutions $\boldsymbol{x} \in \text{conv}(\mathcal{X})$ for which $\tilde{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} = \tilde{\boldsymbol{\theta}}^\top \boldsymbol{x}$. In other words, $\tilde{\boldsymbol{\theta}}^\top \boldsymbol{y} < \tilde{\boldsymbol{\theta}}^\top \boldsymbol{x}\ \forall \boldsymbol{y} \in \text{conv}(\mathcal{X}) \setminus \mathcal{L}$.

We now show that $\mathcal{L} \notin \mathcal{X}^{\text{LR}}(\mathcal{W})$. This is straightforward, as $\boldsymbol{w}^\top \bar{\boldsymbol{x}} > \tau$ and $\boldsymbol{w}^\top \hat{\boldsymbol{x}} > \tau$, which implies that $\boldsymbol{w}^\top \boldsymbol{y} > \tau$ for all $\boldsymbol{y} \in \mathcal{L}$. This implies that $\tilde{\boldsymbol{\theta}}^\top (\hat{\boldsymbol{x}} - \boldsymbol{x}) > 0\ \forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})$, which by definition implies that $\tilde{\boldsymbol{\theta}}$ is in the interior of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. Since $\tilde{\boldsymbol{\theta}}$ is on the boundary of $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ but is in the interior of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$, it must be true that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \not\subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$.

This contradicts the statement that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})) \subseteq \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, which implies that $\mathcal{X}^{\text{LR}}(\mathcal{W})$ cannot be a mixed-RLR. □

A1.4. Proof of Proposition 2

First, note that constraint (14c) excludes only the solution $\hat{\boldsymbol{x}} \in \mathcal{X}$ from $\mathcal{X}$, as it is a reformulation of the no-good cut $|\boldsymbol{x} - \hat{\boldsymbol{x}}|_1 \geq 1$ for binary programs (D'Ambrosio et al. 2010). Second, since $\mathcal{X} = \text{ext}(\mathcal{X})$ when $\mathcal{X} \subseteq \{0,1\}^n$, solving Problem (14) is equivalent to finding the maximizer of $\boldsymbol{w}^\top \boldsymbol{x}$ where $\boldsymbol{x} \in \text{ext}(\mathcal{X}) \backslash \hat{\boldsymbol{x}}$. Let $\tilde{\boldsymbol{x}}$ denote this maximizer, i.e., $\tau^*(\boldsymbol{w}) := \boldsymbol{w}^\top \tilde{\boldsymbol{x}}$. If $\tau^*(\boldsymbol{w}) < \boldsymbol{w}^\top \hat{\boldsymbol{x}}$, then $\hat{\boldsymbol{x}} \in \mathcal{X}$ must be the unique optimal solution under $\boldsymbol{w}$, and $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau^*(\boldsymbol{w})$ would give a tight separating inequality. Otherwise, $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau^*(\boldsymbol{w})$ cannot form a separating hyperplane as $\tau^*(\boldsymbol{w}) \geq \boldsymbol{w}^\top \hat{\boldsymbol{x}}$; instead, it generates a valid inequality of the linear relaxation. This concludes the proof. □

A1.5. Proof of Theorem 3

($\Rightarrow$) First, we prove that if $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$, then there exists a $\boldsymbol{x} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{w}^\top \boldsymbol{x} > \tau$. We prove this by contradiction. Specifically, suppose that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$, but that $\text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\} \subseteq \text{ext}(\mathcal{X}^{\text{LR}}(\mathcal{W}))$. Through the following set of reformulations, we prove that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \supseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$, which is a contradiction:

$$
\begin{aligned}
\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) &= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x},\ \forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}\right\} && \text{(A3a)}\\
&= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x},\ \forall \boldsymbol{x} \in \text{ext}(\mathcal{X}^{\text{LR}})\right\} && \text{(A3b)}\\
&= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x},\ \forall \boldsymbol{x} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}\right\} && \text{(A3c)}\\
&\supseteq \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x},\ \forall \boldsymbol{x} \in \text{ext}(\mathcal{X}^{\text{LR}}(\mathcal{W}))\right\} && \text{(A3d)}\\
&= \left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x},\ \forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})\right\} && \text{(A3e)}\\
&= \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W})). && \text{(A3f)}
\end{aligned}
$$

Equation (A3b) follows from (A3a) by noting that (A3a) can be equivalently rewritten as

$$\left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \max\{\boldsymbol{\theta}^\top \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{X}^{\text{LR}}\}\right\}.$$

Since $\max\{\boldsymbol{\theta}^\top \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{X}^{\text{LR}}\}$ is a linear program, this is also equivalent to

$$\left\{\boldsymbol{\theta} \,\middle|\, \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \max\{\boldsymbol{\theta}^\top \boldsymbol{x} \,|\, \boldsymbol{x} \in \text{ext}(\mathcal{X}^{\text{LR}})\}\right\},$$

from which equation (A3b) is derived.

Equation (A3c) holds since $\boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \hat{\boldsymbol{x}}$ is always satisfied. Equation (A3d) holds from the assumption that $\text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\} \subseteq \text{ext}(\mathcal{X}^{\text{LR}}(\mathcal{W}))$, i.e., the assumption we made to set up the proof by contradiction. Finally, equation (A3e) follows from the same line of reasoning used for (A3b). Since we have proved that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \supseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$, this contradicts our assumption, and the original statement must be true.

($\Leftarrow$) We now show that if there exists a $\tilde{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{w}^\top \tilde{\boldsymbol{x}} > \tau$, then it must be true that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. Our proof focuses on the case where $\hat{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}})$. The case where $\hat{\boldsymbol{x}} \notin \text{ext}(\mathcal{X}^{\text{LR}})$ is easier to prove: if $\hat{\boldsymbol{x}} \notin \text{ext}(\mathcal{X}^{\text{LR}})$, then for any inverse-feasible vector $\boldsymbol{w}$, there must exist $\tilde{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}})$ such that $\boldsymbol{w}^\top \tilde{\boldsymbol{x}} \geq \boldsymbol{w}^\top \hat{\boldsymbol{x}}$. Thus, this would be an example of a $\tilde{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{w}^\top \tilde{\boldsymbol{x}} > \tau$. Furthermore, since $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})) < n$ when $\hat{\boldsymbol{x}} \notin \text{ext}(\mathcal{X}^{\text{LR}})$, and since $\dim(\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))) = n$, it must be true that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$.

Thus, we focus on the case where $\hat{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}})$, and our proof unfolds in three parts:

Part 1 We first show that if there exists a $\tilde{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{w}^\top \tilde{\boldsymbol{x}} > \tau$, then there must also exist a $\bar{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ that is adjacent to $\hat{\boldsymbol{x}}$ that also satisfies $\boldsymbol{w}^\top \bar{\boldsymbol{x}} > \tau$.

Part 2 We then show that, for this $\bar{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$, there exists a $\hat{\boldsymbol{\theta}}$ such that $\bar{\boldsymbol{x}}$ is the only solution to $\max\{\hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \,|\, \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}\}$ where $\hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} = \hat{\boldsymbol{\theta}}^\top \bar{\boldsymbol{x}}$. We show that this $\hat{\boldsymbol{\theta}}$, by definition, must be on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$.

Part 3 Finally, we show that this particular $\hat{\boldsymbol{\theta}}$ is in the interior of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. This implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$.

We now provide the proof for each of these parts.

Proof of Part 1: Let $\widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$ denote the extreme points of $\mathcal{X}^{\text{LR}}$ that are adjacent to $\hat{\boldsymbol{x}}$, i.e., where for any $\boldsymbol{x} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$, the convex combination of $\boldsymbol{x}$ and $\hat{\boldsymbol{x}}$ must lie on the boundary of $\mathcal{X}^{\text{LR}}$. If $\tilde{\boldsymbol{x}} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$, then we are done. Thus, we need to prove that when $\tilde{\boldsymbol{x}} \notin \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$, there does in fact exist $\bar{\boldsymbol{x}} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$ for which $\boldsymbol{w}^\top \bar{\boldsymbol{x}} > \tau$.

We prove this by contradiction. Specifically, we assume that $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau \ \forall \boldsymbol{x} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$; we refer to this as *Assumption A1*. We begin by outlining the steps that lead to a contradiction, after which we prove the key step in detail:

1. Since $\tilde{\boldsymbol{x}} \in \mathcal{X}^{\text{LR}}$ and $\boldsymbol{w}^\top \tilde{\boldsymbol{x}} > \tau$, it must be true that $\tilde{\boldsymbol{x}} \in \mathcal{X}^{\text{LR}} \cap \{\boldsymbol{x} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \geq \tau\}$.
2. We will show that this implies that $\tilde{\boldsymbol{x}} \in \text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\}) \cap \{\boldsymbol{x} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \geq \tau\}$.
3. This implies that $\tilde{\boldsymbol{x}} \in \text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\})$.
4. Since $\tilde{\boldsymbol{x}} \notin \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$ and $\tilde{\boldsymbol{x}} \neq \hat{\boldsymbol{x}}$, this implies that $\tilde{\boldsymbol{x}}$ can be written strictly as a convex combination of other solutions in $\mathcal{X}^{\text{LR}}$, which contradicts the condition that $\tilde{\boldsymbol{x}}$ is an extreme point of $\mathcal{X}^{\text{LR}}$.

The key step in this proof is Step 2. To show this, we prove that when $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau \ \forall \boldsymbol{x} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$ (i.e., Assumption A1), it must be true that

$$\text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\}) \cap \{\boldsymbol{x} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \geq \tau\} = \mathcal{X}^{\text{LR}} \cap \{\boldsymbol{x} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \geq \tau\}.$$

Note that the subset direction ($\subseteq$) must be true, as $\text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\}) \subseteq \mathcal{X}^{\text{LR}}$. Thus, what remains is to prove the superset direction ($\supseteq$). For simplicity, let $\mathcal{G} := \{\boldsymbol{x} \,|\, \boldsymbol{w}^\top \boldsymbol{x} \geq \tau\}$.

To prove this, we will show that

$$\text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\}) \cap \mathcal{G} = \widehat{\text{cone}}(\mathcal{X}^{\text{LR}}) \cap \mathcal{G}, \tag{A4}$$

where

$$\widehat{\text{cone}}(\mathcal{X}^{\text{LR}}) = \{\hat{\boldsymbol{x}} + \sum_{\boldsymbol{x} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})} \lambda_{\boldsymbol{x}}(\boldsymbol{x} - \hat{\boldsymbol{x}}) \,|\, \boldsymbol{\lambda} \geq \boldsymbol{0}\}$$

defines the cone generated by the extreme rays $\{\boldsymbol{y}(\lambda, \boldsymbol{x}, \hat{\boldsymbol{x}})\}_{\lambda \geq 0}\ \forall \boldsymbol{x} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$, where $\boldsymbol{y}(\lambda, \hat{\boldsymbol{x}}, \boldsymbol{x}) := \hat{\boldsymbol{x}} + \lambda(\boldsymbol{y} - \hat{\boldsymbol{x}})$. Once we prove (A4), Step 2 must hold as $\mathcal{X}^{\text{LR}} \subseteq \widehat{\text{cone}}(\mathcal{X}^{\text{LR}})$ (Bodur et al. 2022).

We prove this by construction. For each $\boldsymbol{x}_i \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$, let $\boldsymbol{t}^i$ denote the intersections between the extreme ray $\{\boldsymbol{y}(\lambda, \boldsymbol{x}_i, \hat{\boldsymbol{x}})\}_{\lambda \geq 0}$ and $\boldsymbol{w}^\top \boldsymbol{x} = \tau$. Note that by definition, $\widehat{\text{cone}}(\mathcal{X}^{\text{LR}}) \cap \mathcal{G} = \text{conv}(\boldsymbol{t}^1, \ldots, \boldsymbol{t}^m, \hat{\boldsymbol{x}})$. From Assumption A1, $\boldsymbol{w}^\top \boldsymbol{x}^i < \tau$ and $\boldsymbol{w}^\top \hat{\boldsymbol{x}} > \tau$, which implies that this intersection must have happened on the line segment $\mathcal{L}(\boldsymbol{x}^i, \hat{\boldsymbol{x}}) := \{\boldsymbol{x} \,|\, \rho \boldsymbol{x}^i + (1-\rho)\hat{\boldsymbol{x}},\ \rho \in [0,1]\}$, each of which is feasible in $\text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\})$. Therefore, each point $\boldsymbol{t}^1, \ldots, \boldsymbol{t}^m$ must also be in $\text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\}) \cap \mathcal{G}$, which implies that $\widehat{\text{cone}}(\mathcal{X}^{\text{LR}}) \cap \mathcal{G} \subseteq \text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\}) \cap \mathcal{G}$. Finally, we also know that $\text{conv}(\widehat{\text{ext}}(\mathcal{X}^{\text{LR}}) \cup \{\hat{\boldsymbol{x}}\}) \subseteq \mathcal{X}^{\text{LR}} \subseteq \widehat{\text{cone}}(\mathcal{X}^{\text{LR}})$. Thus, we have proved (A4), and Step 2; we are done with Part 1.

Proof of Part 2: Let $\bar{\boldsymbol{x}} \in \widehat{\text{ext}}(\mathcal{X}^{\text{LR}})$ be the adjacent extreme point to $\hat{\boldsymbol{x}}$ in $\mathcal{X}^{\text{LR}}$ that satisfies $\boldsymbol{w}^\top \bar{\boldsymbol{x}} > \tau$, as proved in Part 1. By definition, $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$ are defined by constraints that differ by only one active constraint, which must imply that there exists linearly independent $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{n-1}$ in $\mathbf{A}$ for which $\boldsymbol{a}_i^\top \hat{\boldsymbol{x}} = \boldsymbol{a}_i^\top \bar{\boldsymbol{x}} = b_i$ for all $i \in \{1, \ldots, n-1\}$; recall we assumed that $\mathcal{X}^{\text{LR}}$ is bounded (Bertsimas and Tsitsiklis 1997). Let $\hat{\boldsymbol{\theta}} = \sum_{i=1}^{n-1} \lambda_i \boldsymbol{a}_i$ where $\lambda_i \in (0,1)$ for all $i \in \{1, \cdots, n-1\}$. By definition, $\hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} = \hat{\boldsymbol{\theta}}^\top \bar{\boldsymbol{x}}$ must hold and $\bar{\boldsymbol{x}}$ is the only solution in $\text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ that satisfies the equality. Furthermore, since $\hat{\boldsymbol{\theta}}^\top (\hat{\boldsymbol{x}} - \bar{\boldsymbol{x}}) = 0$, one of the constraints of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ is active, and $\hat{\boldsymbol{\theta}}$ must be on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$.

Proof of Part 3: We first show that there cannot exist any $\boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})$ for which $\hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} = \hat{\boldsymbol{\theta}}^\top \boldsymbol{x}$. To do so, we define the line segment between $\hat{\boldsymbol{x}}$ and $\bar{\boldsymbol{x}}$ as $\mathcal{L} := \{\boldsymbol{y} \,|\, \boldsymbol{y} = \rho \hat{\boldsymbol{x}} + (1-\rho)(\bar{\boldsymbol{x}}),\ \rho \in [0,1]\}$. Note that by the definition of $\hat{\boldsymbol{\theta}}$, $\mathcal{L} \subseteq \mathcal{X}^{\text{LR}}$ describes the complete set of solutions $\boldsymbol{x} \in \mathcal{X}^{\text{LR}}$ for which $\hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} = \hat{\boldsymbol{\theta}}^\top \boldsymbol{x}$. In other words, $\hat{\boldsymbol{\theta}}^\top \boldsymbol{y} < \hat{\boldsymbol{\theta}}^\top \boldsymbol{x}\ \forall \boldsymbol{y} \in \mathcal{X}^{\text{LR}} \setminus \mathcal{L}$.

We now show that $\mathcal{L} \notin \mathcal{X}^{\text{LR}}(\mathcal{W})$. This is straightforward, as $\boldsymbol{w}^\top \bar{\boldsymbol{x}} > \tau$ and $\boldsymbol{w}^\top \hat{\boldsymbol{x}} > \tau$, which implies that $\boldsymbol{w}^\top \boldsymbol{y} > \tau$ for all $\boldsymbol{y} \in \mathcal{L}$. This implies that $\hat{\boldsymbol{\theta}}^\top (\hat{\boldsymbol{x}} - \boldsymbol{x}) > 0\ \forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W})$, which by definition implies that $\hat{\boldsymbol{\theta}}$ is in the interior of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. Since $\boldsymbol{\theta}$ is on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$ but is not for $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$, it must be true that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. □

A1.6. Proof of Proposition 3

When $\boldsymbol{w}$ is on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}})$, it implies that there exists a $\bar{\boldsymbol{x}} \in \text{ext}(\mathcal{X}^{\text{LR}}) \setminus \{\hat{\boldsymbol{x}}\}$ such that $\boldsymbol{w}^\top \hat{\boldsymbol{x}} = \boldsymbol{w}^\top \bar{\boldsymbol{x}}$. Furthermore, if $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau$ is a separating inequality, then $\boldsymbol{w}^\top \hat{\boldsymbol{x}} = \boldsymbol{w}^\top \bar{\boldsymbol{x}} > \tau$, which implies that $\bar{\boldsymbol{x}} \notin \mathcal{X}^{\text{LR}}(\mathcal{W})$. Theorem 3 implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}))$. □

A1.7. Proof of Theorem 4

We first provide a general overview of this proof, before giving the details. First, we note that $\mathcal{X}_1^{\text{LR}} := \text{conv}(\mathcal{X}^{\text{LR}}(\mathcal{W}_1) \cup \{\hat{\boldsymbol{x}}\})$ is also a valid linear relaxation for $\mathcal{X}$, since all of the extreme points of $\text{conv}(\mathcal{X})$ lie inside the set $\mathcal{X}_1^{\text{LR}}$. Next, note that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1)) = \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}_1^{\text{LR}})$. This is easy to see when we write out their definitions, i.e., note that

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1)) := \{\boldsymbol{\theta} \,|\boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \boldsymbol{\theta}^\top \boldsymbol{x} \;\forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W}_1)\}$$

and

$$\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}_1^{\text{LR}}) := \{\boldsymbol{\theta} \,|\boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq \lambda \boldsymbol{\theta}^\top \hat{\boldsymbol{x}} + (1-\lambda)\boldsymbol{\theta}^\top \boldsymbol{x} \;\forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W}_1), \lambda \in [0,1]\}.$$

Through simple algebra, we observe that the constraints in $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}_1^{\text{LR}})$ are equivalent to

$$(1-\lambda)\boldsymbol{\theta}^\top \hat{\boldsymbol{x}} \geq (1-\lambda)\boldsymbol{\theta}^\top \boldsymbol{x} \;\forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W}_1).$$

Since every inequality in $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$ appears in $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}_1^{\text{LR}})$, and every inequality in $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}_1^{\text{LR}})$ appears in $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$ (by multiplying both sides by a positive constant $\frac{1}{1-\lambda}$ when $\lambda \in [0,1)$), the two sets must be equal.

Using these observations, we provide the proof of the theorem in three steps:

- Step 1: By definition, there must be an extreme point $\boldsymbol{x}^e \in \text{ext}(\mathcal{X}^{\text{LR}}) \backslash \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{x}^e \notin \mathcal{X}^{\text{LR}}(\mathcal{W}_1)$. We will show that there exists a $\boldsymbol{\theta}$ on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$ such that $\boldsymbol{\theta}^\top \boldsymbol{x}^e > \boldsymbol{\theta}^\top \hat{\boldsymbol{x}}$.
- Step 2: By definition, for this $\boldsymbol{\theta}$ there exists an extreme point $\boldsymbol{x}' \in \text{ext}(\mathcal{X}_1^{\text{LR}}) \backslash \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{\theta}^\top \boldsymbol{x}' = \boldsymbol{\theta}^\top \hat{\boldsymbol{x}}$. We will show that $\boldsymbol{x}'$ must lie on the plane $\boldsymbol{w}^\top \boldsymbol{x} = \tau_1$, i.e., $\boldsymbol{w}^\top \boldsymbol{x}' = \tau_1$.
- Step 3: Since $\boldsymbol{w}^\top \boldsymbol{x}' = \tau_1$, this implies that $\boldsymbol{w}^\top \boldsymbol{x}' > \tau_2$ and that $\boldsymbol{x}' \notin \mathcal{X}^{\text{LR}}(\mathcal{W}_2)$. Furthermore, since $\mathcal{X}_1^{\text{LR}}$ defines a valid linear relaxation for $\mathcal{X}$, and since $\boldsymbol{x}' \in \text{ext}(\mathcal{X}_1^{\text{LR}})$, we can apply Theorem 3 to $\mathcal{X}_1^{\text{LR}}$. This implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1)) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_2))$.

We now provide the detailed proofs of each step leading to the final result.

Proof of Step 1: Consider the cone pointed at $\hat{\boldsymbol{x}}$:

$$\text{cone}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1)) := \left\{ \hat{\boldsymbol{x}} + \sum_{\boldsymbol{x} \in \text{ext}(\mathcal{X}^{\text{LR}}(\mathcal{W}_1))} \lambda_{\boldsymbol{x}}(\boldsymbol{x} - \hat{\boldsymbol{x}}) \,\middle|\, \boldsymbol{\lambda} \geq \boldsymbol{0} \right\}. \tag{A5}$$

Let $\boldsymbol{x}^e \in \text{ext}(\mathcal{X}^{\text{LR}}) \backslash \{\hat{\boldsymbol{x}}\}$ be an extreme point where $\boldsymbol{w}^\top \boldsymbol{x}^e > \tau_1$, which implies that $\boldsymbol{x}^e \notin \mathcal{X}^{\text{LR}}(\mathcal{W}_1)$. Furthermore, since the entire line segment $\mathcal{L}(\hat{\boldsymbol{x}}, \boldsymbol{x}^e) := \{\rho \hat{\boldsymbol{x}} + (1-\rho)\boldsymbol{x}^e \,|\, \rho \in [0,1]\}$ is not in $\mathcal{X}^{\text{LR}}(\mathcal{W}_1)$, i.e., $\mathcal{L}(\hat{\boldsymbol{x}}, \boldsymbol{x}^e) \cap \mathcal{X}^{\text{LR}}(\mathcal{W}_1) = \emptyset$, then $\boldsymbol{x}^e \notin \text{cone}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$. Let $\boldsymbol{D}(\boldsymbol{x} - \hat{\boldsymbol{x}}) \leq \boldsymbol{0}$ denote the facets that define $\text{cone}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$, and note that each row $\boldsymbol{d}_i \in \boldsymbol{D}$ must be a boundary point of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$; namely, $\boldsymbol{d}_i^\top \hat{\boldsymbol{x}} \geq \boldsymbol{d}_i^\top \boldsymbol{x} \;\forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W}_1)$, and there exists $\boldsymbol{x}' \in \mathcal{X}^{\text{LR}}(\mathcal{W}_1)$ such that $\boldsymbol{d}_i^\top \hat{\boldsymbol{x}} = \boldsymbol{d}_i \boldsymbol{x}'$. By definition, there must exist a facet $\boldsymbol{d}_i^\top \boldsymbol{x} \leq \boldsymbol{d}_i^\top \hat{\boldsymbol{x}}$ such that $\boldsymbol{d}_i^\top \boldsymbol{x}^e > \boldsymbol{d}_i^\top \hat{\boldsymbol{x}}$; otherwise

$\boldsymbol{x}^e \in \text{cone}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$. For this facet $\boldsymbol{d}_i^\top \boldsymbol{x} \leq \boldsymbol{d}_i^\top \hat{\boldsymbol{x}}$, the vector $\boldsymbol{d}_i$ is an example of a $\boldsymbol{\theta}$ which is on the boundary of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$, and for which $\boldsymbol{\theta}^\top \boldsymbol{x}^e > \boldsymbol{\theta}^\top \hat{\boldsymbol{x}}$.

Proof of Step 2: From Step 1, let $\boldsymbol{x}' \in \text{ext}(\mathcal{X}_1^{\text{LR}}) \backslash \{\hat{\boldsymbol{x}}\}$ be any extreme point of $\mathcal{X}_1^{\text{LR}}$ where $\boldsymbol{\theta}^\top \boldsymbol{x}' = \boldsymbol{\theta}^\top \hat{\boldsymbol{x}}$. We consider a proof by contradiction. Namely, suppose that $\boldsymbol{w}^\top \boldsymbol{x}' < \tau_1$. Let $\boldsymbol{x}^e \in \text{ext}(\mathcal{X}^{\text{LR}}) \backslash \{\hat{\boldsymbol{x}}\}$ denote the extreme point that is cut off by the separating inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau_1$, i.e., $\boldsymbol{w}^\top \boldsymbol{x}^e > \tau_1$. Since $\boldsymbol{x}^e, \boldsymbol{x}' \in \mathcal{X}^{\text{LR}}$, this implies that the line segment $\mathcal{L}(\boldsymbol{x}^e, \boldsymbol{x}') := \{\rho \boldsymbol{x}^e + (1-\rho)\boldsymbol{x}' \,|\, \rho \in [0,1]\}$ lies within $\mathcal{X}^{\text{LR}}$, and more importantly, that there exists a $\rho^* \in (0,1)$ for which $\boldsymbol{w}^\top(\rho^* \boldsymbol{x}^e + (1-\rho^*)\boldsymbol{x}') = \tau_1$. This implies that $(\rho^* \boldsymbol{x}^e + (1-\rho^*)\boldsymbol{x}') \in \mathcal{X}^{\text{LR}}(\mathcal{W}_1)$. However, note that

$$\boldsymbol{\theta}^\top(\rho^* \boldsymbol{x}^e + (1-\rho^*)\boldsymbol{x}') > \boldsymbol{\theta}^\top(\rho^* \hat{\boldsymbol{x}} + (1-\rho^*)\boldsymbol{x}') = \boldsymbol{\theta}^\top \hat{\boldsymbol{x}},$$

which implies that $\boldsymbol{\theta}$ cannot be inverse-feasible. However, since $\boldsymbol{\theta} \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1)) \subset \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$, this must be a contradiction. Thus, $\boldsymbol{x}'$ must satisfy $\boldsymbol{w}^\top \boldsymbol{x}' = \tau_1$.

Proof of Step 3: The proof of Step 3 is straightforward, and follows exactly the procedure outlined in the summary above. Specifically, we can treat $\mathcal{X}_1^{\text{LR}}$ as the existing linear relaxation of $\mathcal{X}$, and we know $\mathcal{X}^{\text{LR}}(\mathcal{W}_2)$ as the RLR generated by adding the separating inequality $\boldsymbol{w}^\top \boldsymbol{x} \leq \tau_2$ to $\mathcal{X}_1^{\text{LR}}$. Since there exists a $\boldsymbol{x}' \in \text{ext}(\mathcal{X}_1^{\text{LR}}) \backslash \{\hat{\boldsymbol{x}}\}$ where $\boldsymbol{w}^\top \boldsymbol{x}' > \tau_2$, this implies that $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}_1^{\text{LR}}) = \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1) \subset \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_2)$. This concludes the proof. □

A1.8. Proof of Theorem 5

The proof of this theorem follows closely from the proofs of Theorems 3 and 4. First, recall that $\mathcal{X}_1^{\text{LR}} := \text{conv}(\mathcal{X}^{\text{LR}}(\mathcal{W}_1) \cup \{\hat{\boldsymbol{x}}\})$ defines a valid linear relaxation of $\mathcal{X}$, and, furthermore, $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}_1^{\text{LR}}) = \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}_1))$; this was shown in the proof of Theorem 4. Now, since $\mathcal{X}_1^{\text{LR}}$ is a valid linear relaxation, we can simply replace $\mathcal{X}^{\text{LR}}$ in Theorem 3 with $\mathcal{X}_1^{\text{LR}}$, and we obtain precisely the result outlined in Theorem 5. □

A1.9. Proof of Proposition 4

Recall that at each outer iteration of Algorithm 1, the master problem (IO-RLR) solves the problem

$$\begin{aligned}
\min_{\boldsymbol{\theta}, \boldsymbol{s}} \quad & f(\boldsymbol{s}) && \text{(A6a)}\\
\text{s.t.} \quad & \boldsymbol{\theta} \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k)) && \text{(A6b)}\\
& \boldsymbol{\theta} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) && \text{(A6c)}\\
& \boldsymbol{\theta} \in \boldsymbol{\Theta}(\boldsymbol{s}) && \text{(A6d)}\\
& \boldsymbol{s} \in \mathcal{S}. && \text{(A6e)}
\end{aligned}$$

Let $(\boldsymbol{\theta}^k, \boldsymbol{s}^k)$ denote the optimal solution to this problem. Since $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k)) \subseteq \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^{k-1}))$ $\forall k \geq 0$, it must be true that $f(\boldsymbol{s}^k) \leq f(\boldsymbol{s}^{k-1})$. It thus suffices that show that if $f(\boldsymbol{s}^k) = f(\boldsymbol{s}^{k-1})$, then $(\boldsymbol{\theta}^k, \boldsymbol{s}^k)$ must be an optimal solution to (IO).

To prove the above statement, we consider a proof by contradiction. Suppose that $f(\boldsymbol{s}^k) = f(\boldsymbol{s}^{k-1})$, but that $f(\boldsymbol{s}^{k-1}) > f(\boldsymbol{s}^*)$ where $\boldsymbol{s}^*$ is the optimal solution to the inverse problem. Recall that $\boldsymbol{\Theta}(\boldsymbol{s})$ defines a linear mapping from $\boldsymbol{s}$ to $\boldsymbol{\theta}$; we will write this as $\boldsymbol{\theta} = \boldsymbol{H}\boldsymbol{s} + \boldsymbol{d}$ where $\boldsymbol{s} \in \mathbb{R}^m$. We prove that this is a contradiction in two steps.

Step 1: We first remark that $\boldsymbol{\theta}^{k-1} := \boldsymbol{H}\boldsymbol{s}^{k-1} + \boldsymbol{d}$ must be in the interior of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k))$. Specifically, $\mathcal{X}^{\text{LR}}(\mathcal{W}^k)$ must include a restricting inequality $(\boldsymbol{\theta}^{k-1})^\top \boldsymbol{x} \leq \tau$, where $\tau := (\boldsymbol{\theta}^{k-1})^\top \hat{\boldsymbol{x}} - \eta$ for some depth parameter $\eta > 0$. This implies that $(\boldsymbol{\theta}^{k-1})^\top \hat{\boldsymbol{x}} > (\boldsymbol{\theta}^{k-1})^\top \boldsymbol{x} \quad \forall \boldsymbol{x} \in \mathcal{X}^{\text{LR}}(\mathcal{W}_k)$, which, by definition, implies that $\boldsymbol{\theta}^{k-1}$ must be in the interior of $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k))$. This last observation is meaningful because it implies that for any bounded vector $\hat{\boldsymbol{s}} \in \mathbb{R}^m$, there exists $\hat{\lambda} > 0$ such that the vector $\boldsymbol{\theta}^{k-1} + \lambda(\boldsymbol{H}\hat{\boldsymbol{s}}) := \boldsymbol{H}(\boldsymbol{s}^{k-1} + \lambda\hat{\boldsymbol{s}}) + \boldsymbol{d}$ remains within $\mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k))$ for all $\lambda \in [0, \hat{\lambda}]$. We will use this fact at the end of Step 2.

Step 2: We now show that if $f(\boldsymbol{s}^{k-1}) > f(\boldsymbol{s}^*)$, then there must exist some $\lambda \in (0, 1]$ for which $\tilde{\boldsymbol{s}} := (1-\lambda)\boldsymbol{s}^{k-1} + \lambda\boldsymbol{s}^*$ is also feasible to the problem (A6). Note that since $f(\boldsymbol{s})$ is a convex function,

$$f(\tilde{\boldsymbol{s}}) = f((1-\lambda)\boldsymbol{s}^{k-1} + \lambda\boldsymbol{s}^*) \leq (1-\lambda)f(\boldsymbol{s}^{k-1}) + \lambda f(\boldsymbol{s}^*) < f(\boldsymbol{s}^{k-1}).$$

Thus, if we can show the existence of such a feasible $\tilde{\boldsymbol{s}}$, then $\boldsymbol{s}^k$ cannot be the optimal solution to (A6), which is a contradiction.

We now show the existence of such a feasible $\tilde{\boldsymbol{s}}$. First, we note that if both $\boldsymbol{s}^{k-1}$ and $\boldsymbol{s}^*$ map to inverse-feasible vectors, then it must be true that its convex combination also maps to inverse-feasible vectors, i.e., $\boldsymbol{H}(\rho\boldsymbol{s}^* + (1-\rho(\boldsymbol{s}^{k-1})) + \boldsymbol{d} = \rho\boldsymbol{\theta}^* + (1-\rho)\boldsymbol{\theta}^{k-1} \in \mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X}) \ \forall \rho \in [0, 1]$; this is true because $\mathcal{R}_{\text{inv}}(\hat{\boldsymbol{x}}, \mathcal{X})$ is a cone. This observation implies that any convex combination of $\boldsymbol{s}^{k-1}$ and $\boldsymbol{s}^*$ must satisfy constraint (A6c). Next, since $\mathcal{S}$ is convex, any convex combination of $\boldsymbol{s}^{k-1} \in \mathcal{S}$ and $\boldsymbol{s}^* \in \mathcal{S}$ must also satisfy (A6e). The only remaining constraint to discuss is (A6b). We now use the result from Step 1, where, for $\hat{\boldsymbol{s}} := \boldsymbol{s}^* - \boldsymbol{s}^{k-1}$, that there must be a feasible $\lambda \in (0, 1]$ such that $\boldsymbol{H}(\boldsymbol{s}^{k-1} + \lambda\hat{\boldsymbol{s}}) + \boldsymbol{d} \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}^{\text{LR}}(\mathcal{W}^k))$. If we define $\tilde{\boldsymbol{s}} := \boldsymbol{s}^{k-1} + \lambda\hat{\boldsymbol{s}} = (1-\lambda)\boldsymbol{s}^{k-1} + \lambda\boldsymbol{s}^*$, then we have found a $\tilde{\boldsymbol{s}}$ that is feasible to (A6) and where $f(\tilde{\boldsymbol{s}}) < f(\boldsymbol{s}^k)$. Thus, $\boldsymbol{s}^k$ cannot be optimal to (A6), which is a contradiction. Algorithm 1 must therefore terminate at the optimal solution. $\square$

A2. Additional Material for Section 4

A2.1. Computing separating inequalities for integer programs

Proposition A1. *Given $\boldsymbol{w} \in \mathbb{R}^n$ and $\hat{\boldsymbol{x}} \in \mathcal{X}$ where $\mathcal{X} \subset \mathbb{Z}^n$ is bounded, the problem*

$$\max_{\boldsymbol{x}} \ \left\{ \boldsymbol{w}^\top \boldsymbol{x} \,\middle|\, \boldsymbol{x} \in \mathcal{X} \backslash \{\hat{\boldsymbol{x}}\} \right\}$$

can be solved as

$$\begin{aligned}
\max_{\boldsymbol{x},\boldsymbol{z},\boldsymbol{\delta}} \quad & \boldsymbol{w}^\top \boldsymbol{x} && \text{(A7a)}\\
\text{s.t.} \quad & \boldsymbol{x} \in \mathcal{X} && \text{(A7b)}\\
& z_i \le x_i - \hat{x}_i + M\delta_i \quad \forall i \in \{1,\dots,n\} && \text{(A7c)}\\
& z_i \le \hat{x}_i - x_i + M(1-\delta_i) \quad \forall i \in \{1,\dots,n\} && \text{(A7d)}\\
& \sum_{i=1}^n z_i \ge 1 && \text{(A7e)}\\
& \boldsymbol{z} \ge \boldsymbol{0},\ \boldsymbol{\delta} \in \{0,1\}^n && \text{(A7f)}
\end{aligned}$$

where M denotes a large constant for which $|\boldsymbol{x}^1 - \boldsymbol{x}^2| \le M,\ \forall \boldsymbol{x}^1, \boldsymbol{x}^2 \in \mathcal{X}$.

Proof of Proposition A1. When $\mathcal{X} \subseteq \mathbb{Z}^n$ is bounded, the set $\mathcal{X}\backslash\{\hat{\boldsymbol{x}}\}$ can written as

$$\mathcal{S}_A := \left\{ \boldsymbol{x} \in \mathcal{X} \,\middle|\, \sum_{i=1}^n |x_i - \hat{x}_i| \ge 1 \right\}, \tag{A8a}$$

since any $\boldsymbol{x} \in \mathcal{X}\backslash\{\hat{\boldsymbol{x}}\}$ must satisfy $|\boldsymbol{x} - \hat{\boldsymbol{x}}|_1 \ge 1$. $\mathcal{S}_A$ is also equivalent to

$$\mathcal{S}_B := \left\{ \boldsymbol{x} \in \mathcal{X} \,\middle|\, \sum_{i=1}^n z_i \ge 1,\ 0 \le z_i \le |x_i - \hat{x}_i| \quad \forall i \in \{1,\dots,n\} \right\}. \tag{A9a}$$

It is easy to verify that for any $\boldsymbol{x} \in \mathcal{S}_A$, there exists a $\boldsymbol{z}$ for which $(\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{S}_B$, and similarly, for any $(\boldsymbol{x}, \boldsymbol{z}) \in \mathcal{S}_B$, the value of $\boldsymbol{x}$ must satisfy $\boldsymbol{x} \in \mathcal{S}_A$. Furthermore, $\mathcal{S}_B$ is by definition equivalent to

$$\mathcal{S}_C := \left\{ \boldsymbol{x} \in \mathcal{X} \,\middle|\, \sum_{i=1}^n z_i \ge 1,\ 0 \le z_i \le \max\{x_i - \hat{x}_i, \hat{x}_i - x_i\} \quad \forall i \in \{1,\dots,n\} \right\}. \tag{A10a}$$

In $\mathcal{S}_C$, the constraint

$$0 \le z_i \le \max\{x_i - \hat{x}_i, \hat{x}_i - x_i\} \quad \forall i \in \{1,\dots,n\}$$

can be reformulated as

$$\begin{aligned}
& z_i \le x_i - \hat{x}_i + M\delta_i \quad \forall i \in \{1,\dots,n\}\\
& z_i \le \hat{x}_i - x_i + M(1-\delta_i) \quad \forall i \in \{1,\dots,n\}\\
& z_i \ge 0 \quad \forall i \in \{1,\dots,n\}\\
& \delta_i \in \{0,1\} \quad \forall i \in \{1,\dots,n\}.
\end{aligned}$$

We note that this is a valid reformulation by considering all three cases: (i) when $x_i - \hat{x}_i = 0$, $z_i = 0$; (ii) when $x_i - \hat{x}_i \ge 1$, $\delta_i = 0$ and $z_i \le x_i - \hat{x}_i$; (iii) when $\hat{x}_i - x_i \ge 1$, $\delta_i = 1$ and $z_i \le \hat{x}_i - x_i$. $\square$

A2.2. Details of Example 3

This section provides additional details on Example 3. To generate the approximations of the inverse-feasible region, we consider $\mathcal{K}(\hat{\boldsymbol{x}}, \tilde{\mathcal{X}})$ where $\tilde{\mathcal{X}}$ is defined by the standard linear relaxation (LR), a tighter linear relaxation (LR$^+$), and the four different RLRs detailed in Example 3:

- **LR** ($\mathcal{X}^{\text{LR}}$): $\{\boldsymbol{x} \,|\, \mathbf{A}\boldsymbol{x} \leq \boldsymbol{b},\ \boldsymbol{x} \in [0,1]^n\}$
- **LR$^+$**: $\{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}}\}$
- **RLR0.5**: $\{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} - 0.5\eta^*\}$
- **RLR1**: $\{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} - \eta^*\}$
- **RLR2**: $\{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} - 2\eta^*\}$
- **RLR3**: $\{\boldsymbol{x} \in \mathcal{X}^{\text{LR}} \,|\, \hat{\boldsymbol{\theta}}^\top \boldsymbol{x} \leq \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} - 3\eta^*\}$

As mentioned in the example, the value $\eta^* := \hat{\boldsymbol{\theta}}^\top \hat{\boldsymbol{x}} - \tau^*(\hat{\boldsymbol{\theta}})$ in Figure 4 denotes the depth of the tight separating inequality generated by solving Problem (14) with $\boldsymbol{w} = \hat{\boldsymbol{\theta}}$. By definition, **RLR0.5** and **RLR1** define pure-RLRs, whereas **RLR2** and **RLR3** define mixed-RLRs.

Table A1 provides the sensitivity ranges of each objective coefficient θ_i. The leftmost column provides the nominal values of $\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{x}}$, whereas the rightmost column provides the true sensitivity ranges. The remaining columns provide the ranges generated by the linear relaxations and RLRs.

Table A1: Sensitivity ranges for all θ_i for different linear relaxations

index	$\hat{\theta}_i$	$\hat{x}_i$	**LR**		**LR$^+$**		**RLR0.5**		**RLR1**		**RLR2**		**RLR3**		**TRUE**	
			Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+
0	560	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	43
1	1125	0	-	-	$-\infty$	0	$-\infty$	7.11	$-\infty$	14	$-\infty$	27.17	$-\infty$	39.59	$-\infty$	189
2	300	0	-	-	$-\infty$	0	$-\infty$	7.26	$-\infty$	13.53	$-\infty$	26	$-\infty$	39	$-\infty$	62
3	620	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-74	∞
4	2100	0	-	-	$-\infty$	0	$-\infty$	28.57	$-\infty$	53.22	$-\infty$	93.77	$-\infty$	125.68	$-\infty$	809
5	431	1	-	-	0	∞	-11.72	∞	-23.44	∞	-41	∞	-54.66	∞	-140	∞
6	68	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	19
7	328	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-85	∞
8	47	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-27	∞
9	122	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	33
10	322	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-87	∞
11	196	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-19	∞
12	41	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-32	∞
13	25	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	13
14	425	1	-	-	0	∞	-18.87	∞	-35.86	∞	-62.57	∞	-83.27	∞	-248	∞
15	4260	1	-	-	0	∞	-26.63	∞	-53.26	∞	-98.37	∞	-137.19	∞	-1389	∞
16	416	1	-	-	0	∞	-19.45	∞	-38.90	∞	-67.87	∞	-90.32	∞	-245	∞

Continued on next page

Table A1 – continued from previous page

index	$\hat{\theta}_i$	$\hat{x}_i$	**LR**		**LR$^+$**		**RLR0.5**		**RLR**		**RLR** (-2η)		**RLR** (-3η)		**TRUE**	
			Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+	Δ_i^-	Δ_i^+
17	115	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	41
18	82	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-44	∞
19	22	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-13	∞
20	631	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	72
21	132	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	57
22	420	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-76	∞
23	86	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	39
24	42	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-13	∞
25	103	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-19	∞
26	215	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-75	∞
27	81	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-29	∞
28	91	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-71	∞
29	26	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	13
30	49	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-43	∞
31	420	1	-	-	0	∞	-18.36	∞	-36.72	∞	-64.07	∞	-85.23	∞	-285	∞
32	316	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	18
33	72	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-31	∞
34	71	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-13	∞
35	49	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-19	∞
36	108	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-58	∞
37	116	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-19	∞
38	90	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-64	∞
39	738	1	-	-	0	∞	-30.79	∞	-61.57	∞	-107.54	∞	-143.17	∞	-382	∞
40	1811	1	-	-	0	∞	-65	∞	-130	∞	-226.81	∞	-301.70	∞	-884	∞
41	430	1	-	-	0	∞	-19.39	∞	-38.77	∞	-67.63	∞	-89.96	∞	-286	∞
42	3060	1	-	-	0	∞	-148.41	∞	-296.83	∞	-518.42	∞	-690.16	∞	-2076	∞
43	215	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-38	∞
44	58	0	-	-	$-\infty$	0	$-\infty$	6.50	$-\infty$	13	$-\infty$	26	$-\infty$	39	$-\infty$	13
45	296	0	-	-	$-\infty$	0	$-\infty$	8.09	$-\infty$	16.18	$-\infty$	28.26	$-\infty$	39	$-\infty$	77
46	620	1	-	-	0	∞	-14.73	∞	-29.45	∞	-51.54	∞	-68.77	∞	-181	∞
47	418	1	-	-	0	∞	-10.83	∞	-21.65	∞	-37.89	∞	-50.53	∞	-190	∞
48	47	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-18	∞
49	81	1	-	-	0	∞	-6.50	∞	-13	∞	-26	∞	-39	∞	-61	∞

Note that the two linear relaxations **LR** and **LR$^+$** are entirely uninformative when used for sensitivity analysis. For **LR**, we are not able to generate any ranges on $\hat{\boldsymbol{\theta}}$ simply because $\hat{\boldsymbol{\theta}}$ does not render $\hat{\boldsymbol{x}}$ optimal over the relaxation of Problem (11). For the tighter linear relaxation **LR$^+$**, $\hat{\boldsymbol{\theta}}$ defines a normal vector of a facet tangent to $\hat{\boldsymbol{x}}$, and any perturbations on $\hat{\boldsymbol{\theta}}$ in a non-trivial

direction (i.e., negative direction when $x_i = 1$ or positive direction when $x_i = 0$) will not retain the optimality of $\hat{\boldsymbol{x}}$ over this relaxation. Figure A1 provides an illustration.

(a) (b)

Figure A1 **An illustration of why $\mathcal{X}^{\text{LR}}$ and $\mathcal{X}^{\text{LR}+}$ may not lead to informative sensitivity ranges. Specifically, any perturbation of $\hat{\boldsymbol{\theta}}$ in a non-trivial direction (downwards) will no longer render $\hat{\boldsymbol{x}}$ optimal over either relaxations.**

We now outline how we calculated the average coverage and level of overapproximation that is shown in Figure 4. The percentage of coverage of sensitivity ranges for each **RLRK** where $\mathbf{K} \in \{0.5, 1, 2, 3\}$ is calculated using the formula

$$\frac{\sum_{i \in n} (\Delta_i^{+(\mathbf{TRUE})} - \Delta_i^{+(\mathbf{RLRK})}) + (\Delta_i^{-(\mathbf{TRUE})} - \Delta_i^{-(\mathbf{RLRK})})}{\sum_{i \in n} (\Delta_i^{+(\mathbf{TRUE})} + \Delta_i^{-(\mathbf{TRUE})})}$$

where any value of ∞ or $-\infty$ is replaced with the value of 0, as one direction will always be unbounded when $x_i \in \{0, 1\}$ is a binary variable. Similarly, the percentages of overapproximation can be calculated using the formula

$$\frac{\sum_{i \in n} v_i^k}{\sum_{i \in n} (\Delta_i^{+(\mathbf{TRUE})} + \Delta_i^{-(\mathbf{TRUE})})}$$

where each v_i^k tabulates the overapproximation (if any) of **RLRK** for each coordinate $i \in n$, i.e.,

$$v_i^k = \begin{cases} \Delta_i^{+(\mathbf{RLRK})} - \Delta_i^{+(\mathbf{TRUE})} & \text{if} \quad \Delta_i^{+(\mathbf{RLRK})} - \Delta_i^{+(\mathbf{TRUE})} > 0 \\ \Delta_i^{-(\mathbf{RLRK})} - \Delta_i^{-(\mathbf{TRUE})} & \text{if} \quad \Delta_i^{-(\mathbf{RLRK})} - \Delta_i^{-(\mathbf{TRUE})} > 0 \\ 0 & \text{otherwise.} \end{cases}$$

A3. Additional Material for Section 5

A3.1. Details of the adaptive policy

With a poorly specified static depth selection policy, the RLR-CP Algorithm may either degenerate into the standard cutting plane algorithm when the depth of restricting inequalities are too large, or become overly conservative when the depths are too shallow. We propose an adaptive policy that allows for real-time modifications to the depth parameters chosen by the static policy. This can both mitigate the algorithm's sensitivity to the static policy and enhance its overall performance.

In its most general form, the adaptive policy adjusts the depth of the current or subsequent restricting inequality based on the time spent in the cutting plane subroutine. Specifically, as more time is spent within the subroutine, the depth of the most recently added restricting inequality can be progressively decreased. Conversely, if the subroutine is largely bypassed – indicating that the depth is too conservative – the next restricting inequality that is added can be initialized with a larger depth value. To implement this adaptive policy, we require introducing an additional t index to η^k, where η_t^k now denotes the depth of the k-th restricting inequality at iteration t in the cutting plane subroutine.

To address the first case, we can define an adaptive policy that decreases the depth parameter of the most recently added restricting inequality after each (or several) iteration(s) of this subroutine. One simple example would be to adjust η_t^k as

$$\eta_t^k = \frac{\eta_0^k}{t},$$

where the depth of the restricting inequality $(\boldsymbol{\theta}^k)^\top \boldsymbol{x} \leq (\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}} - \eta_t^k$ is iteratively made more shallow as cuts are generated by the cutting plane subroutine.

To address the second case, we can define an adaptive policy that modifies the baseline depth of the next restricting inequality based on the number of iterations spent in the cutting plane subroutine under the current restricting inequality. Specifically, recall from equation (19) that a fixed $\tilde{\eta}$ is used to define a static policy where $\eta_{t=0}^{k+1} = \tilde{\eta}|(\boldsymbol{\theta}^{k+1})^\top \hat{\boldsymbol{x}}|$. To implement an adaptive policy, we can simply replace $\tilde{\eta}$ with $\tilde{\eta}^{k+1}$ where $\tilde{\eta}^{k+1}$ is a decreasing function in the number of cuts added by the subroutine in iteration k, which we denote as T^k. For example, we can define $\tilde{\eta}^k$ as

$$\tilde{\eta}^{k+1} = \frac{\alpha}{T^k}\tilde{\eta}^k$$

for some value of $\alpha > 1$.

We conclude by noting that the main purpose of this section is to provide a conceptual framework for defining this adaptive policy, and there may be many alternative specifications for how we can adjust the depth of restricting inequalities based on real-time performance.

A4. Additional Material for Section 6.2

A4.1. Generation of forward problem instances

The multi-objective knapsack problems that we consider in the numerical experiments can be written as

$$\max_{\boldsymbol{x}} \left\{ \sum_{j=1}^{10} s_i \boldsymbol{f}_i \boldsymbol{x} \;\middle|\; \boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b},\, \boldsymbol{x} \in \{0,1\}^n \right\}$$

where $n = \{100, 200, 400, 800\}$. For each n, the knapsack problem is defined using a constraint matrix $\mathbf{A} \in \mathbb{R}^{5\times n}$ consisting of randomly generated integers between 0 and 201, a vector $\boldsymbol{b}$ that is generated by taking 40% of the sum of entries of each row of $\mathbf{A}$, and vectors $\boldsymbol{f}_i$ that are composed of randomly generated integers between 50 and 200.

A4.2. Details of CP algorithms

We consider two cutting plane algorithms as benchmarks, one of which serves as the cutting plane subroutine in our RLR-CP algorithm. Note that the main difference between these benchmarks and our RLR-CP algorithm is that the benchmarks simply do not contain constraint $\boldsymbol{\theta} \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}(\mathcal{W}^k))$. Specifically, the standard CP algorithm (Wang 2009) is defined precisely by the subroutine given in IO-RLR of Algorithm 1 when we remove the constraint $\boldsymbol{\theta} \in \mathcal{K}(\hat{\boldsymbol{x}}, \mathcal{X}(\mathcal{W}^k))$ from CP-M. Similarly, the trust-region based CP algorithm (Bodur et al. 2022) differs only in the definition of IO-RLR; instead of solving

$$\max\{(\tilde{\boldsymbol{\theta}}^t)^\top \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{X}\}, \tag{A11}$$

at each iteration, the trust-region based CP algorithm solves

$$\max\{(\tilde{\boldsymbol{\theta}}^t)^\top \boldsymbol{x} \,|\, \boldsymbol{x} \in \mathcal{X} \cap \mathrm{TR}(\hat{\boldsymbol{x}}, \delta)\} \tag{A12}$$

where $\mathrm{TR}(\hat{\boldsymbol{x}}, \delta) = \{\boldsymbol{x} \,|\, \|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_1 \leq \delta\}$ and δ controls the size of the L_1-norm ball centered at $\hat{\boldsymbol{x}}$.

We implemented the same version of the trust-region based CP algorithm that was implemented in Bodur et al. (2022): starting with $\delta^{(0)} = 1$ where $k = 0$, we solve the CP subroutine where δ in (A12) is replaced with $\delta^{(k)} = 2^k$. When cuts can no longer be generated using $\delta^{(k)}$ (i.e., there does not exist any $\boldsymbol{x}^t \in \mathcal{X} \cap \mathrm{TR}(\hat{\boldsymbol{x}}, \delta^{(k)})$ such that $(\tilde{\boldsymbol{\theta}}^t)^\top \boldsymbol{x}^t > (\tilde{\boldsymbol{\theta}}^t)^\top \hat{\boldsymbol{x}}$), we increase the value of k to $k+1$, doubling the value of δ. Then, for every 10 cuts that are generated, we force $\delta = \infty$ for one iteration, which enables the algorithm to terminate if it cannot find any violated cut in all of $\mathcal{X}$.

Finally, to speed up both the benchmark CP algorithms and the CP subroutines, we consider the early-stop heuristic from Bodur et al. (2022), where, rather than solving (A11) and (A12) to optimality at each iteration, we can instead stop the solver earlier when two conditions are both satisfied: (i) at least 5 seconds have elapsed and (ii) the best solution found satisfies the condition $(\boldsymbol{\theta}^k)^\top \boldsymbol{x}^t > (\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}}$. This early-stop heuristic enables a more efficient cut generation process for both the CP and RLR-CP algorithms.

A4.3. Additional results and insights

In Section 6, we examined the aggregated results for the multi-objective knapsack problems with 200 and 400 variables. For completeness, we provide the results for the 100 and 800 variable problems in Tables A2 and A3. We note that the three key insights outlined in Section 6.2 remain consistent across these additional instances.

Next, we provide further analysis on the depth parameters used in the numerical experiments. Specifically, we compare the value of the best performing depth parameters to those that would have been generated if the algorithm had only computed tight separating inequalities. The purpose

Time Limit	**5s**				**10s**				**20s**			
Optimality Gap	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**
RLR–CP: $\tilde{\eta} = 0.0001$	0	0	0	0	0	0	0	0	0	0	0	0
RLR–CP: $\tilde{\eta} = 0.001$	7	2	0	0	10	9	6	2	10	10	10	9
RLR–CP: $\tilde{\eta} = 0.01$	6	5	5	5	7	7	7	7	10	10	10	10
RLR–CP: $\tilde{\eta} = 0.1$	1	1	1	1	5	5	5	5	8	8	8	8
CP: Standard	0	0	0	0	4	4	4	4	7	7	7	7
CP: Trust Region	1	1	1	1	4	4	4	4	8	8	8	8

Table A2 **Counts of 100 variable instances reaching specified optimality gaps within time limits.**

Time Limit	**500s**				**1500s**				**3000s**			
Optimality Gap	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**	**15%**	**5%**	**1%**	**0%**
RLR–CP: $\tilde{\eta} = 0.00001$	7	4	2	0	10	6	5	0	10	10	5	0
RLR–CP: $\tilde{\eta} = 0.0001$	6	3	1	0	7	6	3	2	8	7	5	2
RLR–CP: $\tilde{\eta} = 0.001$	0	0	0	0	6	6	3	0	9	9	6	2
RLR–CP: $\tilde{\eta} = 0.01$	0	0	0	0	3	3	2	0	7	7	6	2
RLR–CP: $\tilde{\eta} = 0.1$	0	0	0	0	3	3	1	0	8	8	3	1
CP: Standard	0	0	0	0	0	0	0	0	8	8	8	8
CP: Trust Region	0	0	0	0	2	2	2	2	8	8	8	8

Table A3 **Counts of 800 variable instances reaching specified optimality gaps within time limits.**

of this comparison is to understand whether the insights illustrated in Example 3 hold, i.e., whether generating separating inequalities may be overly conservative.

To derive these insights, we apply Algorithm 1 to the same IO instances by generating only separating inequalities. Specifically, given any candidate $\boldsymbol{\theta}^k$ generated at an iteration k in this new implementation, we solve Problem (14) in RLR-UPDATE to generate a tight separating inequality $(\boldsymbol{\theta}^k)^\top \boldsymbol{x} \leq \tau^*(\boldsymbol{\theta}^k)$, stopping only when it is no longer possible to do so, i.e., when $(\boldsymbol{\theta}^{\tilde{k}})^\top \hat{\boldsymbol{x}} = \tau^*(\boldsymbol{\theta}^{\tilde{k}})$ for some iteration $\tilde{k}$.

	100 Vars.	**200 Vars.**	**400 Vars.**	**800 Vars.**
Avg. absolute depth (η^k)	2.25	2.02	1.43	0.2
Avg. relative depth ($\eta^k / \lvert(\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}}\rvert$)	0.00034	0.00016	0.00005	0.000004

Table A4 **Average depth parameter values for $k < \tilde{k}$, averaged across 10 instances for model size.**

Table A4 shows the depth parameter values, averaged across all iterations $k < \tilde{k}$ and all problem instances for the different variable sizes. The first row shows the average depth in absolute values, while the second row shows the average depth in terms of the multiplicative value $\tilde{\eta}^k$ for which $\tilde{\eta}^k \lvert(\boldsymbol{\theta}^k)^\top \hat{\boldsymbol{x}}\rvert$ would have to led to this same value. If we compare the values of the second row to

those corresponding to the best average-performing choice of $\tilde{\eta}$ in Tables A2 (100 variables), 2 (200 variables), 3 (400 variables), and A3 (800 variables), we find that the best performing choices of $\tilde{\eta}$ are larger than the values in the second row. To pick out one example, we can consider the 200 variable instances, shown in Table 2. In these instance, setting a depth parameter of $\tilde{\eta} = 0.0001$, which is almost identical to the corresponding value in the second row of Table A4, leads to an algorithm that performs much worse than if $\tilde{\eta} = 0.001$. This insight is consistent with what was observed in Example 3, i.e., that separating inequalities may be overly conservative and that deeper restricting inequalities – albeit not excessively deep – can cover significantly more of the inverse-feasible region and result in better performing algorithms. When we increase $\tilde{\eta}$ even further to $\tilde{\eta} = 0.01$, the performance of the algorithm degrades again. Similar insights can be observed across all instances. In summary, these observation reinforce the idea that the best performing specification of Algorithm 1 are those that strike a balance between using RLRs and the cutting plane subroutine, rather than trying to avoid the latter altogether.