Adjustable robust optimization with objective uncertainty

Boris Detienne[a], Henri Lefebvre[b], Enrico Malaguti[b], Michele Monaci[b]

[a]*IMB UMR CNRS 5251, Inria Bordeaux Sud-Ouest, Université de Bordeaux, 200 Avenue de la Vieille Tour, Talence, 33405, France*
[b]*Dipartimento di Ingegneria dell'Energia Elettrica e dell'Informazione "Guglielmo Marconi", Università di Bologna, Viale del Risorgimento, 2, Bologna, 40136, Italy*

Abstract

In this work, we study optimization problems where some cost parameters are not known at decision time and the decision flow is modeled as a two-stage process within a robust optimization setting. We address general problems in which all constraints (including those linking the first and the second stages) are defined by convex functions and involve mixed-integer variables, thus extending the existing literature to a much wider class of problems. We show how these problems can be reformulated using Fenchel duality, allowing to derive an enumerative exact algorithm, for which we prove asymptotic convergence in the general case, and finite convergence for cases where the first-stage variables are all integer.

An implementation of the resulting algorithm, embedding a column generation scheme, is then computationally evaluated on a variant of the Capacitated Facility Location Problem with uncertain transportation costs, using instances that are derived from the existing literature. To the best of our knowledge, this is the first approach providing results on the practical solution of this class of problems.

Keywords:
Uncertainty modelling, two-stage robust optimization, reformulation, Fenchel duality, branch-and-price, computational experiments

Email addresses: `boris.detienne@u-bordeaux.fr` (Boris Detienne), `henri.lefebvre@unibo.it` (Henri Lefebvre), `enrico.malaguti@unibo.it` (Enrico Malaguti), `michele.monaci@unibo.it` (Michele Monaci)

1. Introduction

Robust Optimization (RO) has emerged as a solution approach to deal with uncertainty in optimization problems. Contrary to stochastic optimization, another popular appraoch, robust optimization does not rely on probability distributions. Indeed, RO considers an uncertainty set for the unknown parameters, against which the taken decision should be immune. In that sense, constraints have to be respected in every possible realization of the parameters and the objective function evaluated in the least advantageous case. The concept was first introduced in Soyster (1973) and received considerable attention in the scientific literature. Recent advances in RO can be found in Bertsimas et al (2010), Hassene et al (2009), Ben-Tal et al (2009), Leyffer et al (2020) and Yanıkoğlu et al (2019), among others.

More formally, a basic (one-stage) robust optimization problem can be cast as follows

$$\begin{aligned} \inf_{\boldsymbol{z}} \quad & \sup_{\boldsymbol{\xi}\in\Xi} f(\boldsymbol{\xi},\boldsymbol{z}) \\ \text{subject to} \quad & \boldsymbol{g}(\boldsymbol{\xi},\boldsymbol{z}) \leq \boldsymbol{0} \qquad \forall \boldsymbol{\xi}\in\Xi \\ & \boldsymbol{z}\in Z. \end{aligned} \tag{1SR-P}$$

Here, the unknown data is represented by variables $\boldsymbol{\xi}$ that belong to the so-called *uncertainty set* Ξ. As mentioned above, decision $\boldsymbol{z}$ has to be feasible in every possible occurrence of the uncertainty, hence robust solutions tend to be overly conservative. To tackle this drawback, Ben-Tal et al (2004) introduced the so-called *adjustable robust optimization*, also known as *two-stage robust optimization*. As its name suggests, in a two-stage context, part of the decisions are made in a *first stage* (i.e., here-and-now, before uncertainty reveals), while recourse decisions can be taken in a *second stage* (i.e., once the actual values of the uncertain data are known) as an attempt to react to the outcome of the uncertain process. Typically, the feasible region of (1SR-P) can, indeed, be recast to embed a two-stage decision process by splitting variables $\boldsymbol{z}$ in $(\boldsymbol{x},\boldsymbol{y})$. Here, $\boldsymbol{x}\in\mathcal{X}$ are decisions to be made here and now, while $\boldsymbol{y}\in\mathcal{Y}$ may be taken at a later instant. Accordingly, set Z is defined as $\mathcal{X}\times\mathcal{Y}$. With the convention that the minimum objective function value for an infeasible problem is $+\infty$, a two-stage robust problem can be formulated as follows

$$\inf_{\boldsymbol{x}\in\mathcal{X}} \sup_{\boldsymbol{\xi}\in\Xi} \inf_{\boldsymbol{y}\in\mathcal{Y}(\boldsymbol{x},\boldsymbol{\xi})} f(\boldsymbol{\xi},\boldsymbol{x},\boldsymbol{y}), \tag{2SR-P}$$

where $\mathcal{Y}(\boldsymbol{x},\boldsymbol{\xi}) = \{\boldsymbol{y} : \boldsymbol{y} \in \mathcal{Y},\ \boldsymbol{g}(\boldsymbol{\xi},\boldsymbol{x},\boldsymbol{y}) \leq \boldsymbol{0}\}$, and $\boldsymbol{g}(\boldsymbol{\xi},\boldsymbol{x},\boldsymbol{y}) \leq \boldsymbol{0}$ are the so-called *linking constraints*. Set $\mathcal{X}$ is now referred to as the first-stage feasible region. Given $\overline{\boldsymbol{x}} \in \mathcal{X}$ and $\overline{\boldsymbol{\xi}} \in \Xi$, the resulting second-stage feasible region is $\mathcal{Y}(\overline{\boldsymbol{x}},\overline{\boldsymbol{\xi}})$, and the second-stage problem is $\inf\{f(\overline{\boldsymbol{\xi}},\overline{\boldsymbol{x}},\boldsymbol{y}) : \boldsymbol{y} \in \mathcal{Y}(\overline{\boldsymbol{x}},\overline{\boldsymbol{\xi}})\}$. It is known (see, Ben-Tal et al (2004)) that most of the problems which can be cast as two-stage robust problems are at least NP-hard. This result even holds for cases where first and second-stage variables are continuous and all the involved functions are linear. Several approaches have been developed to tackle this class of problems. Assuming that the second stage is continuous and exhibits strong duality, it can be replaced by its dual. This way, the inner maximization problem can be reformulated using its epigraph, leading to a constraint-generation algorithm in the spirit of Benders' decomposition (see, e.g., Terry et al (2009), Bertsimas et al (2013), Jiang et al (2014) and Gabrel et al (2011)). A column-and-constraint-generation scheme has been proposed in Zeng and Zhao (2013), which consists in adding one set of second-stage decision variables and constraints associated with each realization of the uncertainty. These realizations are dynamically generated by solving a so-called adversarial problem which identifies the worst-case scenario for a current estimate of the first-stage decisions. The algorithm stops when no such scenario can be found. Later, the same approach was used in Ayoub and Poss (2016), where the authors model the adversarial problem as a mixed integer program, derived using Farkas' lemma and standard linearization techniques. Note that the column-and-constraint generation approach can handle mixed-integer second-stage decisions, which is not the case for classical Benders-type approaches. Unfortunately, this method seems to be of practical relevance only when a small number of variables has to be added for reaching optimality.

The inherent difficulty of this class of problems motivated the development of approximate solution methods. In the *affine decision rule* approach (Ben-Tal et al (2004)), the second-stage decisions are expressed as affine functions of the uncertainty. Another relevant approach, introduced in Bertsimas and Caramanis (2010), is the *finite adaptability* (also known as K-adaptability) in which the number of second-stage decisions is restricted to some finite number. An MILP formulation for the case of binary second-stage decisions and objective uncertainty was proposed in Hanasusanto et al (2015) and a branch-and-bound algorithm was later proposed in Subramanyam et al (2019) to address cases with uncertain linear constraints.

An important special case of (2SR-P) arises when uncertainty affects the

objective function only, i.e., $\mathcal{Y}(\boldsymbol{x}, \boldsymbol{\xi}) = \mathcal{Y}(\boldsymbol{x}), \forall \boldsymbol{\xi} \in \Xi$. For this specific case, Kämmerling and Kurtz (2020) proposed an oracle-based algorithm relying on a hull relaxation combining the first- and second-stage feasible regions embedded within a branch-and-bound framework. However, this approach applies to first-stage binary variables and linear constraints only. On the other hand, Arslan and Detienne (2022) proposed an exact MILP reformulation of the problem in case of linear linking constraints that involve binary variables only. Besides solving the problem by means of a branch-and-price algorithm, a further contribution of Arslan and Detienne (2022) is proving the NP-completeness of the problem in this setting.

Our analysis shows that, in the setting where uncertainty affects the objective function only, no contribution has been presented in the literature for tackling problems where linking constraints are defined by nonlinear functions or involve both integer and continuous variables. Similarly, to the best of our knowledge, the case in which the objective function is nonlinear has not been considered yet. This paper contributes in filling this gap, as we consider two-stage robust problems with objective uncertainty, convex constraints and mixed-integer first and second stage. By extending in a non-trivial way some recent results from the two-stage stochastic optimization literature (see Sherali and Fraticelli (2002), Sherali and Zhu (2006) and Li and Grossmann (2019)), we obtain a relaxation of the problem, and analyze its tightness for different special cases. This relaxation can be embedded within an enumerative scheme thus producing an exact solution approach, for which we prove asymptotic convergence in the general case, and finite convergence in the integer case. Besides the theoretical analysis, we also show that, from a computational viewpoint, the proposed algorithm is able to solve instances of practical relevance arising from the logistic field . We also point out that the class of problems which can be addressed by our solution approach is quite large since we only require mild assumptions on the nature of the involved optimization problem.

The article is organized as follows. In Section 2 we formally introduce the considered class of problems, whereas in Section 3 we present a relaxation of the problem. We then derive sufficient conditions for the relaxation to coincide with the original problem in a mixed-integer context. So as to close the optimality gap, we introduce an enumerative algorithm which embeds a spatial branching mechanism on continuous first-stage variables. We prove asymptotic convergence of the overall algorithm in presence of continuous first-stage decisions and finite ε-convergence in case of integer first-stage de-

cisions. In Section 3.4, we propose a column-generation algorithm to solve the relaxation problem. Finally, Section 4 applies the proposed method to a capacitated facility location problem with congestion.

Notations Throughout this paper, matrices and vectors are written in bold case, e.g., $\boldsymbol{x} \in \mathbb{R}^n$ or $\boldsymbol{A} \in \mathbb{R}^{n\times m}$, while components are written in normal font, e.g., x_i or a_{ij}. Columns of $\boldsymbol{A}$ are written in bold case with exponent indexing, e.g., $\boldsymbol{a}^i$. Let $f : \mathbb{R}^n \to \mathbb{R}$ be a given function with $\mathrm{dom}(f) = \{\boldsymbol{x} \in \mathbb{R}^n : f(\boldsymbol{x}) < +\infty\}$; its convex conjugate is denoted by $f^* : \mathbb{R}^n \to \mathbb{R}$ and is given by

$$f^*(\boldsymbol{\pi}) = \sup_{\boldsymbol{x}\in \mathrm{dom}(f)} \{\boldsymbol{\pi}^T\boldsymbol{x} - f(\boldsymbol{x})\}.$$

Similarly, we denote by f_* the concave conjugate of f. Let $X \subseteq \mathbb{R}^n \times \mathbb{Z}^{n-p}$ be a given set, described in terms of constraints and integer restrictions. We denote by $\overline{X}$ its continuous relaxation and by $\mathrm{conv}\,(X)$ its convex hull, i.e., the smallest convex set C satisfying $X \subseteq C$.

The indicator function of X is noted $\delta(\cdot|X)$ and equals zero if its argument belongs to X and $+\infty$ otherwise. Its convex conjugate is therefore given by $\delta^*(\boldsymbol{\pi}|X) = \sup\{\boldsymbol{\pi}^T\boldsymbol{x} : \boldsymbol{x} \in X\}$. Basic results on conjugate calculus are summarized in Appendix A. If X is a convex polytope, we note $\mathrm{vert}\,(X)$ the set of its extreme points. Finally, for a logical proposition $\mathcal{E}$, function $\mathbf{1}(\mathcal{E})$ equals one if $\mathcal{E}$ is true and zero otherwise.

2. Problem description

2.1. General setting

As anticipated, our goal is to solve problem (2SR-P) with objective uncertainty, convex constraints and mixed-integer first and second stages.

For the sake of clarity, let us first introduce several sets. Set $I = \{1, \ldots, n_1\}$ denotes the set of indices for the first-stage variables, partitioned into two sets I_I and I_C. Variables whose index belongs to I_I are required to take integer values, while those whose index belongs to I_C are continuous variables, i.e., wlog, $\mathcal{X} \subset \mathbb{R}^{|I_C|} \times \mathbb{Z}^{|I_I|}$. Similarly, we introduce set $J = \{1, \ldots, n_2\}$ as the indices for the second-stage variables and partition this set into J_I and J_C, i.e., wlog, $\mathcal{Y} \subset \mathbb{R}^{|J_C|} \times \mathbb{Z}^{|J_I|}$. Finally, we introduce set $U = \{1, \ldots, n_3\}$ as the index set for the uncertain variables, i.e., $\Xi \subset \mathbb{R}^{n_3}$.

We now explicit some assumptions on the problem.

Assumption 1 (Objective uncertainty)**.** *For all* $\boldsymbol{\xi} \in \Xi$ *and* $\boldsymbol{x} \in \overline{\mathcal{X}}$, $\mathcal{Y}(\boldsymbol{\xi}, \boldsymbol{x}) = \mathcal{Y}(\boldsymbol{x})$.

Assumption 2 (Convexity)**.**

1. $\overline{\mathcal{X}}$ *is compact and convex;*
2. *The uncertainty set* Ξ *is a finite-dimensional, bounded convex set;*
3. *For all* $\boldsymbol{x} \in \overline{\mathcal{X}}$, $\overline{\mathcal{Y}}(\boldsymbol{x})$ *is a finite-dimensional, bounded convex set;*
4. *The objective function* f *is a concave function of the uncertainty and a convex function of the first- and second-stage decisions, i.e.,* $f_{\boldsymbol{x},\boldsymbol{y}} : \boldsymbol{\xi} \mapsto f(\boldsymbol{\xi}, \boldsymbol{x}, \boldsymbol{y})$ *is a concave function for all fixed* $\boldsymbol{x} \in \overline{\mathcal{X}}$ *and* $\boldsymbol{y} \in \overline{\mathcal{Y}}(\boldsymbol{x})$*, and* $f_{\boldsymbol{\xi}} : (\boldsymbol{x}, \boldsymbol{y}) \mapsto f(\boldsymbol{\xi}, \boldsymbol{x}, \boldsymbol{y})$ *is a convex function for all fixed* $\boldsymbol{\xi} \in \Xi$.

Assumption 3 (Complete recourse)**.** *For every (relaxed) first-stage decision, there exists at least one feasible second-stage decision, i.e., for every* $\boldsymbol{x} \in \overline{\mathcal{X}}$, $\mathcal{Y}(\boldsymbol{x})$ *is a non-empty set.*

Assumption 4 (Boundedness)**.**

1. *The objective function* f *is bounded over the first- and second-stage feasible region, i.e., for all fixed* $\boldsymbol{\xi} \in \Xi$, $\{(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x} \in \overline{\mathcal{X}}, \boldsymbol{y} \in \overline{\mathcal{Y}}(\boldsymbol{x})\} \subseteq \operatorname{dom}(f_{\boldsymbol{\xi}})$*;*
2. *For all* $(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x} \in \overline{\mathcal{X}}$ *and* $\boldsymbol{y} \in \overline{\mathcal{Y}}(\boldsymbol{x})$, $\operatorname{relint}(\Xi) \cap \operatorname{dom}(f_{\boldsymbol{x},\boldsymbol{y}}) \neq \emptyset$.

Assumption 5 (Separability)**.** *Let* $Q = \{1, \ldots, q\}$.

1. *The objective function* f *can be expressed as a sum of* q *functions, i.e., there exist* q *functions* $(\psi_i : \mathbb{R}^{|U|+|I|+|J|} \to \mathbb{R})_{i \in Q}$ *such that* $f(\boldsymbol{\xi}, \boldsymbol{x}, \boldsymbol{y}) = \sum_{i \in Q} \psi_i(\boldsymbol{\xi}, \boldsymbol{x}, \boldsymbol{y})$ *for all* $\boldsymbol{x} \in \mathcal{X}, \boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x})$ *and all* $\boldsymbol{\xi} \in \Xi$*;*
2. *For all* $i \in Q$, ψ_i *is separable in* $\boldsymbol{\xi}$ *and* $(\boldsymbol{x}, \boldsymbol{y})$ *meaning that there exists functions* $(w_i : \mathbb{R}^{|U|} \to \mathbb{R})_{i \in Q}$ *and* $(\varphi_i : \mathbb{R}^{|I|+|J|} \to \mathbb{R})_{i \in Q}$ *such that* $\psi_i(\boldsymbol{\xi}, \boldsymbol{x}, \boldsymbol{y}) = w_i(\boldsymbol{\xi})\varphi_i(\boldsymbol{x}, \boldsymbol{y})$*. In addition, we assume that* $w_i(\cdot)$ *is a concave function and* $\varphi_i(\cdot)$ *is a convex function.*

A few remarks regarding these assumptions are necessary. First, note that Assumptions 1 and 2 are here to define what we refer to as *convex mixed-integer robust problems with objective uncertainty*. We highlight that the word "convex" is here to suggest that all involved functions are convex with respect to the first- and second-stage variables. Yet, in general, even under these assumptions, problem (2SR-P) may fail to have a straightforward convex MINLP formulation. Indeed, function $h : \boldsymbol{x} \mapsto \max_{\boldsymbol{\xi} \in \Xi} \min_{\boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x})} f(\boldsymbol{\xi}, \boldsymbol{x}, \boldsymbol{y})$ is not necessarily a convex function over the continuous relaxation of $\mathcal{X}$. We give here a small example.

Example 1 (nonconvex MINLP)**.** *Consider the following first- and second-stage feasible regions.*

$$\mathcal{X} = [0,1] \text{ and } \mathcal{Y}(x) = \left\{ \boldsymbol{y} \in \{0,1\}^2 \middle| \begin{array}{rl} y_1 + y_2 & \leq 1 \\ y_1 & \leq 1 - x \end{array} \right\}.$$

By inspection, we have that $(y_1, y_2) = (0,0)$ *and* $(y_1, y_2) = (0,1)$ *are always feasible second-stage solutions, while* $(y_1, y_2) = (1,0)$ *is feasible only when* $x = 0$*. Fixing the uncertainty set* $\Xi = [0,1]$*, we take interest in the following* convex mixed-integer two-stage robust problem

$$\min_{x \in [0,1]} h(x) \text{ with } h : x \mapsto \max_{\xi \in [0,1]} \min_{(y_1,y_2) \in \mathcal{Y}(x)} \xi(-2y_1 + y_2 + 1).$$

Though every involved functions are convex (in fact, affine) with respect to $\boldsymbol{\xi}, \boldsymbol{x}$ *and* $\boldsymbol{y}$*, we have that*

$$h(x) = \begin{cases} \max\limits_{\xi \in [0,1]} \min\{\xi; 2\xi; -\xi\} = 0 & \text{if } x = 0 \\ \max\limits_{\xi \in [0,1]} \min\{\xi; 2\xi\} = 1 & \text{if } x > 0 \end{cases} = \mathbf{1}(x > 0).$$

Clearly, h *fails to be convex over* $[0,1]$ *which ends our example.*

Assumption 3 is a standard assumption in the two-stage optimization literature, and is known to be easy to enforce as soon as the considered problem is bounded, which is implied by Assumption 4.1. Assumption 4.2 is not restrictive in practice, and will be used in the proof of Lemma 2.

Finally, Assumption 5 is structural to our work, and implies the following remarks.

Remark 1. The assumption that $\varphi_i(\cdot)$ is a convex function (at most affine) for all $i \in Q$ is without loss of generality.

Proof. Let $i \in Q$ such that $\varphi_i(\cdot)$ is concave, then, to fulfill Assumption 2.4, $w_i(\boldsymbol{\xi})$ must be negative forall $\boldsymbol{\xi} \in \Xi$. Thus, one may equivalently replace $w_i(\cdot)$ by $-w_i(\cdot)$ and $\varphi_i(\cdot)$ by $-\varphi_i(\cdot)$. □

Remark 2. For all $i \in Q$ such that $\varphi_i(\cdot)$ (resp. $w_i(\cdot)$) is not single-signed, then $w_i(\cdot)$ (resp. $\varphi_i(\cdot)$) is affine.

Remark 3. For all $i \in Q$ such that $\varphi_i(\cdot)$ (resp. $w_i(\cdot)$) is not affine, then $w_i(\cdot)$ (resp. $\varphi_i(\cdot)$) is a non-negative function.

Note that Assumption 5 could be relaxed to address situations in which, for $i \in Q$ such that $\varphi_i(\cdot)$ (resp. $w_i(\cdot)$) is affine, there is no restriction on the concavity (resp. convexity) of the associated $w_i(\cdot)$ (resp. $\varphi_i(\cdot)$).

Example 2 (Fulfilling Assumption 5)**.** *We give here some examples of functions which satisfy Assumption 5. For simplicity, we denote* $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y})$.

- *Uncertain linear functions of the form* $(\boldsymbol{\xi}, \boldsymbol{z}) \mapsto \boldsymbol{\xi}\boldsymbol{A}\boldsymbol{z}$ *where* $\boldsymbol{A}$ *is a given real matrix;*
- *Diagonal uncertain convex quadratic form* $(\boldsymbol{\xi}, \boldsymbol{z}) \mapsto \boldsymbol{z}^T diag(\boldsymbol{\xi})\boldsymbol{z}$ *where* $\boldsymbol{\xi} \geq 0$*;*
- *Uncertain positively weighted sum of convex functions of the form* $(\boldsymbol{\xi}, \boldsymbol{z}) \mapsto \sum_{i \in Q} \xi_i \varphi_i(\boldsymbol{z})$ *with* $\Xi \subset \mathbb{R}_+^{|U|}$*, e.g.,* $(\boldsymbol{\xi}, \boldsymbol{x}, \boldsymbol{y}) \mapsto \sum_{i \in Q} \xi_i x_i^2 / y_i$ *with* $\boldsymbol{y} \geq \boldsymbol{0}$.

Example 3 (Violating Assumption 5)**.** *We give here some examples of functions which do not satisfy Assumption 5.*

- *Non-concave functions of the uncertainty, e.g.,* $(\boldsymbol{\xi}, \boldsymbol{z}) \mapsto ||\boldsymbol{z} - \boldsymbol{\xi}||$ *for any given norm;*
- *General uncertain quadratic form* $(\boldsymbol{\Sigma}, \boldsymbol{z}) \mapsto \boldsymbol{z}^T \boldsymbol{\Sigma} \boldsymbol{z}$ *even with* $\boldsymbol{\Sigma} \succeq 0$ *(unless* $\Xi \cap \mathbb{R}_-^{|U|} = \emptyset$*).*

In the following lemma, we finally state the class of problems we consider.

Lemma 1. *Under Assumptions 1-5, there exists* $[\boldsymbol{l}, \boldsymbol{u}] \subset \mathbb{R}^{|I|+|J|}$ *such that* (2SR-P) *is equivalent to the following two-stage optimization problem with convex objective function and objective uncertainty*

$$\inf_{\boldsymbol{x} \in \mathcal{X} \cap [\boldsymbol{l}, \boldsymbol{u}]} \sup_{\boldsymbol{\xi} \in \Xi} \inf_{(\boldsymbol{t}, \boldsymbol{y}) \in \mathcal{Y}'(\boldsymbol{x})} \sum_{i \in Q} w_i(\boldsymbol{\xi}) t_i \tag{2SRO-P}$$

with $\mathcal{Y}'(\boldsymbol{x})$ *such that* $\mathcal{Y}(\boldsymbol{x}) = \text{proj}_{\boldsymbol{y}}(\mathcal{Y}'(\boldsymbol{x}))$ *and* $\overline{\mathcal{Y}}'(\boldsymbol{x})$ *is a convex and finite-dimensional set.*

Proof. The existence of the hyper-rectangle $[\boldsymbol{l}, \boldsymbol{u}]$ is trivial as $\mathcal{X}$ is assumed to be bounded (Assumption 2.1). Moreover, the following equality holds.

$$\inf_{\boldsymbol{y}} \left\{ \sum_{i \in Q} w_i(\boldsymbol{\xi}) \varphi_i(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x}) \right\} = \inf_{\boldsymbol{y}, \boldsymbol{t}} \left\{ \sum_{i \in Q} w_i(\boldsymbol{\xi}) t_i : \boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x}), t_i = \varphi_i(\boldsymbol{x}, \boldsymbol{y}), \forall i \in Q \right\}$$

However, the optimization problem on the right side of the equality may fail to be convex if there exists $i \in Q$ such that φ_i is not affine. Let $Q^A \subseteq Q$ be the set of indices for which φ_i is affine. By Assumption 5, for all $i \in Q \backslash Q^A$, we have $w_i(\cdot) \geq 0$ and thus constraint "$t_i = \varphi_i(\boldsymbol{x}, \boldsymbol{y})$" may be equivalently replaced by "$t_i \geq \varphi_i(\boldsymbol{x}, \boldsymbol{y})$", which is convex. We therefore can choose

$$\mathcal{Y}'(\boldsymbol{x}) = \left\{ (\boldsymbol{t}, \boldsymbol{y}) : \begin{array}{ll} \boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x}) & \\ t_i = \varphi_i(\boldsymbol{x}, \boldsymbol{y}) & \forall i \in Q^A \\ t_i \geq \varphi_i(\boldsymbol{x}, \boldsymbol{y}) & \forall i \in Q \backslash Q^A \end{array} \right\}.$$

For every $\boldsymbol{x} \in \overline{\mathcal{X}}$, the continuous relaxation of $\mathcal{Y}'(\boldsymbol{x})$ is convex and non-empty (Assumption 3); by construction, it is also finite dimensional. □

In what remains, we will assume to know a hyper-rectangle $[\boldsymbol{l}, \boldsymbol{u}]$ as described in Lemma 1.

2.2. Special case: linear and binary setting

We complete the introduction by discussing the special case of (2SRO-P) under the following additional assumptions:

1. $\mathcal{X}, \Xi$ and $\boldsymbol{x} \mapsto \mathcal{Y}(\boldsymbol{x})$ are defined by linear constraints;
2. there exists a matrix $\boldsymbol{A} \in \mathbb{R}^{|U| \times |Q|}$ such that $\forall i \in Q, w_i(\boldsymbol{\xi}) = \boldsymbol{\xi}^T \boldsymbol{a}^i$; and
3. linking constraints are defined by functions $g(\boldsymbol{x}, \boldsymbol{y})$ that do not depend on first-stage variables in I_c.

In a recent paper Arslan and Detienne (2022), the authors observed that, for this variant of the problem, the inner minimization $\min_{\boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x})} \boldsymbol{\xi}^T \boldsymbol{A} \boldsymbol{y}$ can be equivalently replaced by $\min_{\boldsymbol{y} \in \text{conv}(\mathcal{Y}(\boldsymbol{x}))} \boldsymbol{\xi}^T \boldsymbol{A} \boldsymbol{y}$, i.e., the second-stage feasible region can be substituted by its convex hull. This allows to transform the min-max-min problem into a min-max problem by the well known minimax theorem. Assuming that Ξ is expressed as $\{\boldsymbol{\xi} \in \mathbb{R}_+^{|U|} : \boldsymbol{F}\boldsymbol{\xi} \leq \boldsymbol{d}\}$, the inner maximization problem is dualized so as to obtain the following equivalent problem

$$\min_{\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\lambda}} \quad \boldsymbol{d}^T \boldsymbol{\lambda} \tag{1}$$

$$\text{subject to } \boldsymbol{x} \in \mathcal{X} \tag{2}$$

$$\boldsymbol{y} \in \text{conv}\left(\mathcal{Y}(\boldsymbol{x})\right) \tag{3}$$

$$\boldsymbol{F}^T \boldsymbol{\lambda} \geq \boldsymbol{A} \boldsymbol{y} \tag{4}$$

$$\boldsymbol{\lambda} \geq \boldsymbol{0}, \tag{5}$$

where $\boldsymbol{\lambda}$ are the dual variables associated to the inner maximization problem. Note that, besides the integrality requirements on the variables, the only nonconvex constraint is (3). By exploiting a reformulation already used in Sherali and Fraticelli (2002), Sherali and Zhu (2006) and Li and Grossmann (2019) for two-stage stochastic optimization problems with mixed-integer first and second stage, Arslan and Detienne (2022) showed that, for each fixed $\bar{\boldsymbol{x}} \in \mathcal{X}$, $\{\bar{\boldsymbol{x}}\} \times \operatorname{conv}(\mathcal{Y}(\bar{\boldsymbol{x}})) = \operatorname{conv}(S) \cap \{(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) : \boldsymbol{x} = \bar{\boldsymbol{x}}\}$ where $S = \{(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x} \in \{0,1\}, \boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x})\}$. Hence, constraint (3) may be equivalently enforced as

$$(\boldsymbol{x}, \boldsymbol{y}) \in \operatorname{conv}(S). \tag{6}$$

The obtained reformulation is then solved by means of a branch-and-price algorithm where branching is performed on the first-stage variables only.

3. A hull-relaxation-based branch-and-price algorithm

In this section we present our main contribution and its theoretical foundations. We first turn problem (2SRO-P) from a min-max-min problem to a min-max problem in our mixed-integer and convex context. Then, since linear duality does not apply in our setting, we resort to Fenchel duality to obtain a reformulation of the problem. Similarly to the linear and binary case, we then replace the counterpart of (3) by constraints which play the same role as (6). This only provides a relaxation of the problem in the general setting. This relaxation is thus embedded into an enumerative scheme to obtain an optimal solution of (2SRO-P).

3.1. Problem reformulation

The following lemma extends the result given in Arslan and Detienne (2022) to the mixed-integer and convex context.

Lemma 2 (Single-stage reformulation)**.** *Problem* (2SRO-P) *is equivalent to the following problem*

$$\inf_{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y}) \in F} \sup_{\boldsymbol{\xi} \in \Xi} \sum_{i \in Q} w_i(\boldsymbol{\xi}) t_i \tag{7}$$

with $F = \{(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) : \boldsymbol{x} \in \mathcal{X} \cap [\boldsymbol{l}, \boldsymbol{u}], (\boldsymbol{t}, \boldsymbol{y}) \in \operatorname{conv}(\mathcal{Y}'(\boldsymbol{x}))\}$.

Proof. This lemma relies on the same arguments as those employed in Arslan and Detienne (2022): first, the feasible region of the inner minimization problem is replaced by its convex hull. This is valid by linearity of the

objective function and convexity of the feasible region. By Assumption 2.2 and Lemma 2, both Ξ and $\text{conv}\left(\mathcal{Y}'(\boldsymbol{x})\right)$ (for all $\boldsymbol{x} \in \mathcal{X}$) are convex and finite dimensional sets. Thus, the minmax theorem in Perchet and Vigeral (2015) can be used to turn the inner $\sup - \inf$ into an $\inf - \sup$ problem. This achieves the proof. □

The inner maximization problem may be turned into a minimization problem by use of Fenchel duality, as done in Ben-Tal et al (2009). In the following proposition, we therefore derive a general convex reformulation of problem (2SRO-P).

Proposition 1 (Deterministic reformulation)**.** *Problem* (2SRO-P) *is equivalent to the following problem*

$$\inf_{\boldsymbol{x},\boldsymbol{y},\boldsymbol{t},(\boldsymbol{v}^i)_{i\in Q},\boldsymbol{\xi}} \delta^*(\boldsymbol{\xi}|\Xi) - \sum_{i\in Q}(t_i w_i)_*\left(\boldsymbol{v}^i\right) \tag{8}$$

$$\text{subject to } \boldsymbol{x} \in \mathcal{X} \cap [\boldsymbol{l}, \boldsymbol{u}] \tag{9}$$

$$(\boldsymbol{t}, \boldsymbol{y}) \in \text{conv}\left(\mathcal{Y}'(\boldsymbol{x})\right) \tag{10}$$

$$\sum_{i\in Q} \boldsymbol{v}^i = \boldsymbol{\xi} \tag{11}$$

$$\boldsymbol{v}^i \in \mathbb{R}^{|U|} \qquad \forall i \in Q. \tag{12}$$

Proof. By a direct application of Fenchel duality and some conjugate calculus results, the following holds.

$$\begin{aligned}
\sup_{\boldsymbol{\xi}\in\Xi} \sum_{i\in Q} t_i w_i(\boldsymbol{\xi}) &= \sup_{\boldsymbol{\xi}\in\mathbb{R}^{|U|}} \left\{ \sum_{i\in Q} t_i w_i(\boldsymbol{\xi}) - \delta(\boldsymbol{\xi}|\Xi) \right\} = \inf_{\boldsymbol{\xi}\in\mathbb{R}^{|U|}} \left\{ \delta^*(\boldsymbol{\xi}|\Xi) - \left(\sum_{i\in Q} t_i w_i(\boldsymbol{\xi}) \right)_* \right\} \\
&= \inf_{\boldsymbol{\xi}\in\mathbb{R}^{|U|}} \left\{ \delta^*(\boldsymbol{\xi}|\Xi) - \sup_{\boldsymbol{v}^i\in\mathbb{R}^{|U|}, i\in Q} \left\{ \sum_{i\in Q} (t_i w_i)_*\left(\boldsymbol{v}^i\right) : \sum_{i\in Q} \boldsymbol{v}^i = \boldsymbol{\xi} \right\} \right\} \\
&= \inf \left\{ \delta^*(\boldsymbol{\xi}|\Xi) - \sum_{i\in Q} (t_i w_i)_*\left(\boldsymbol{v}^i\right) : \sum_{i\in Q} \boldsymbol{v}^i = \boldsymbol{\xi}, \boldsymbol{v}^i \in \mathbb{R}^{|U|}, i \in Q, \boldsymbol{\xi} \in \mathbb{R}^{|U|} \right\}.
\end{aligned}$$

See also Appendix A for more details on conjugate calculus. □

The following results show that, although the reformulation for the general case adds $|Q| \times |U|$ continuous variables, these additional variables can be omitted for some relevant cases. In particular this is true in case all the $w_i(\cdot)$ functions are either separable or affine.

Remark 4. Assume wlog that $|Q| = |U|$. If, for all $i \in Q$, $w_i(\boldsymbol{\xi}) = w_i(\xi_i)$, then problem (2SRO-P) is equivalent to

$$\inf_{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y})\in F} \left\{ \delta^*(\boldsymbol{\xi}|\Xi) - \sum_{i\in Q} (t_i w_i)_* (\boldsymbol{\xi}) \right\}. \tag{13}$$

Proof. By assumption, we have $(t_i w_i)_*(\boldsymbol{v}^i) = (t_i w_i)_*(v_{ii})$. In addition, (11) is as follows

$$\sum_{i\in Q} v_{ij} = \xi_j \qquad j = 1, ..., |U|, \tag{14}$$

and v_{ij}, $i \neq j$, does not appear in the objective function. Thus, there always exists an optimal solution such that $\xi_j = v_{jj}$, $j = 1, ..., |U|$. □

Remark 5. Let $i \in Q$ such that $w_i(\cdot)$ is affine, i.e., $w_i(\boldsymbol{\xi}) = (\boldsymbol{r}^i)^T \boldsymbol{\xi} + r_{i0}$. Problem (2SRO-P) is equivalent to

$$\inf_{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y})\in F} \left\{ \delta^*(\boldsymbol{R}\boldsymbol{t}|\Xi) + \boldsymbol{r}_0^T \boldsymbol{t} \right\}. \tag{15}$$

Proof. Indeed, we have

$$(t_i w_i)_*(\boldsymbol{v}) = \inf_{\boldsymbol{\xi}\in\mathbb{R}^{|U|}} \{\boldsymbol{v}^T\boldsymbol{\xi} - t_i((\boldsymbol{r}^i)^T\boldsymbol{\xi} + r_{i0})\} = \begin{cases} -t_i r_{i0} & \text{if } \boldsymbol{v} = t_i \boldsymbol{r}^i \\ -\infty & \text{otherwise.} \end{cases}$$

□

3.2. Relaxation

Note that the deterministic reformulation presented above still is not, in general, a convex MINLP. Indeed, $\mathcal{Y}'(\boldsymbol{x})$ in constraints (10) depends on variables $\boldsymbol{x}$. Since no tractable compact form is known in the general case, we replace constraint $(\boldsymbol{t},\boldsymbol{y}) \in \operatorname{conv}(\mathcal{Y}'(\boldsymbol{x}))$ by the following relaxed requirement.

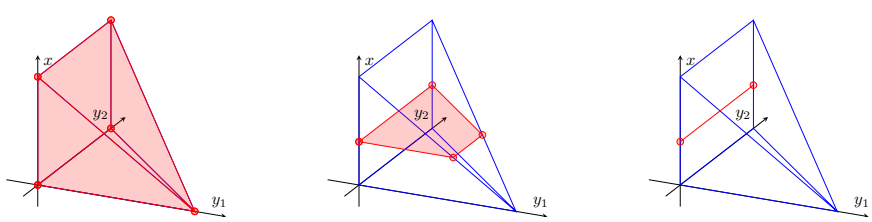
$$(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y}) \in \operatorname{conv}(S) \text{ with } S = \left\{ (\boldsymbol{x},\boldsymbol{t},\boldsymbol{y}) : \begin{array}{ll} l_j \leq x_j \leq u_j & \forall j \in I \\ x_j \in \mathbb{Z} & \forall j \in I_I \\ (\boldsymbol{t},\boldsymbol{y}) \in \mathcal{Y}'(\boldsymbol{x}) & \end{array} \right\}. \tag{16}$$

The substitution yields the following problem, which is a relaxation of (8)–(12)

$$\begin{aligned} \min_{\boldsymbol{x},\boldsymbol{y},(\boldsymbol{v}^i)_{i\in Q},\boldsymbol{\xi}} \quad & \delta^*(\boldsymbol{\xi}|\Xi) - \sum_{i\in Q}(t_i w_i)_*\left(\boldsymbol{v}^i\right) \\ \text{subject to} \quad & \boldsymbol{x} \in \mathcal{X} \cap [\boldsymbol{l},\boldsymbol{u}] \\ & (\boldsymbol{x},\boldsymbol{t},\boldsymbol{y}) \in \operatorname{conv}(S) \\ & \sum_{i\in Q} \boldsymbol{v}^i = \boldsymbol{\xi} \\ & \boldsymbol{v}^i \in \mathbb{R}^{|U|} \qquad \forall i \in Q \\ & \boldsymbol{\xi} \in \mathbb{R}^{|U|}. \end{aligned} \tag{P}$$

It is clear that, for any fixed $\bar{\boldsymbol{x}} \in \mathcal{X}$, we have $\{\bar{\boldsymbol{x}}\} \times \mathcal{Y}'(\bar{\boldsymbol{x}}) = S \cap \{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y}) : \boldsymbol{x} = \bar{\boldsymbol{x}}\}$, and that the same holds even for $\bar{\boldsymbol{x}} \in \overline{\mathcal{X}}$. However, as shown, e.g., in Sherali and Zhu (2006), the convexified counterpart does not hold, in the sense that the inclusion "$\{\bar{\boldsymbol{x}}\} \times \operatorname{conv}(\mathcal{Y}(\bar{\boldsymbol{x}})) \subseteq \operatorname{conv}(S) \cap \{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y}) : \boldsymbol{x} = \bar{\boldsymbol{x}}\}$" may be strict. Example 4 below illustrates this case.

Example 4 (Hull relaxation)**.** *We consider the first- and second-stage feasible sets introduced in Example 1. In Figure* (1a)*, we represent the convex hull of S. For a fixed first-stage decision $\bar{x}$ (here, $\bar{x} = 0.4$), Figure* (1b) *reports the feasible points for constraint* (16)*, whereas Figure* (1c) *describes the exact shape of* $\operatorname{conv}(\mathcal{Y}(\bar{x}))$*. The figure shows an example in which inclusion is strict. In addition, note that, whenever $\bar{x}$ attains its bounds (i.e., $\bar{x} \in \{0,1\}$), $\{\bar{x}\} \times \operatorname{conv}(\mathcal{Y}(\bar{x})) = \operatorname{conv}(S) \cap \{(x,\boldsymbol{y}) : x = \bar{x}\}$ holds.*



(a) $\operatorname{conv}(S)$ (b) $\operatorname{conv}(S) \cap \{x = \bar{x}\}$ (c) $\{\bar{x}\} \times \operatorname{conv}(\mathcal{Y}(\bar{x}))$

Figure 1: Graphical representation of different sets from example 1

The following Lemma follows from the considerations above.

Lemma 3 (Lower-bounding property)**.** *Denoting by $v(\bullet)$ the optimal objective value of problem $\bullet$, we have*

$$v(\text{P}) \leq v(\text{2SRO-P}).$$

In other words, (P) is a relaxation of (2SRO-P). In the next proposition, we introduce a condition under which a feasible solution for problem (P) is feasible for problem (2SRO-P) as well.

Proposition 2. *If* $\bar{\boldsymbol{x}} \in \operatorname{vert}([\boldsymbol{l}, \boldsymbol{u}])$*, then*

$$\{\bar{\boldsymbol{x}}\} \times \operatorname{conv}(\mathcal{Y}'(\bar{\boldsymbol{x}})) = \operatorname{conv}(S) \cap \{(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) : \boldsymbol{x} = \bar{\boldsymbol{x}}\}.$$

Proof. Let $\bar{\boldsymbol{x}} \in \operatorname{vert}([\boldsymbol{l}, \boldsymbol{u}])$ and let $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{t}}, \hat{\boldsymbol{y}}) \in \operatorname{conv}(S) \cap \{(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) : \boldsymbol{x} = \bar{\boldsymbol{x}}\}$. Then, $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{t}}, \hat{\boldsymbol{y}})$ can be expressed as a (finite) convex combination of points of $\operatorname{conv}(S)$ (Carathéodory's theorem), i.e.,

$$(\hat{\boldsymbol{x}}, \hat{\boldsymbol{t}}, \hat{\boldsymbol{y}}) = \sum_{e \in E} (\bar{\mathbf{x}}^e, \bar{\mathbf{t}}^e, \bar{\mathbf{y}}^e) \alpha_e,$$

where E is a given index list of such elements of $\operatorname{conv}(S)$. Assume that there exists $j \in I$ and $i \in E$ such that $\bar{\mathrm{x}}_j^i \neq \bar{x}_j$. If $\bar{\mathrm{x}}_j^i > \bar{x}_j$, condition $\bar{\mathbf{x}}^i \in \operatorname{conv}(S)$ implies that $\bar{x}_j = l_j$. Hence, $\alpha_i = 0$ since $\bar{\mathrm{x}}_j^k \geq l_j \ \forall k \in E$. The same argument shows that $\bar{\mathrm{x}}_j^i < \bar{x}_j$ implies $\alpha_i = 0$. Thus, for each $e \in E$ such that $\alpha_e > 0$, we must have $\bar{\mathbf{x}}^e = \bar{\boldsymbol{x}}$. This implies that $(\bar{\mathbf{t}}^e, \bar{\mathbf{y}}^e) \in \mathcal{Y}'(\bar{\boldsymbol{x}})$ and thus $\sum_{e \in E} (\bar{\mathbf{t}}^e, \bar{\mathbf{y}}^e) \alpha_e \in \operatorname{conv}(\mathcal{Y}'(\bar{\boldsymbol{x}}))$. □

Corollary 1 (Tightness condition)**.** *Let* X^* *be the set of optimal first-stage decisions of problem* (P)*. Then*

$$X^* \cap \operatorname{vert}([\boldsymbol{l}, \boldsymbol{u}]) \neq \emptyset \Rightarrow v(\mathrm{P}) = v(\text{2SRO-P}).$$

Proof. Let $(\boldsymbol{x}^*, \boldsymbol{t}^*, \boldsymbol{y}^*)$ be an optimal solution of (P) with $\boldsymbol{x}^* \in \operatorname{vert}([\boldsymbol{l}, \boldsymbol{u}])$. From Proposition 2, it is also feasible for problem (2SRO-P). Thus, Lemma 3 implies optimality for problem (2SRO-P). □

This result directly implies Corollary 2 which states that, in the special case where the first-stage variables are all binary, problem (P) is always an exact reformulation of (2SRO-P).

Corollary 2 (Tightness condition/binary case)**.** *If the first-stage decisions are all binary, i.e.,* $I_C = \emptyset$ *and* $\forall j \in I_I, l_j = 0, u_j = 1$*, then*

$$v(\mathrm{P}) = v(\text{2SRO-P}).$$

Proof. In this case, $[\boldsymbol{l}, \boldsymbol{u}] = [\mathbf{0}, \mathbf{1}]$, hence any optimal first-stage solution $\boldsymbol{x}^*$ satisfies $\boldsymbol{x}^* \in \{0, 1\}^{|I_I|} = \operatorname{vert}([\boldsymbol{l}, \boldsymbol{u}])$ which, by Corollary 1, proves the result. □

3.3. Enumerative algorithm

We now present an exact method for solving problem (2SRO-P) which works as follows:

- we exploit the deterministic reformulation (8)–(12) of the problem;
- we relax the integrality of the $\boldsymbol{x}$, and impose that $\boldsymbol{x} \in \overline{\mathcal{X}}$;
- we relax requirement (10), i.e., $(\boldsymbol{t}, \boldsymbol{y}) \in \operatorname{conv}(\mathcal{Y}'(\boldsymbol{x}))$, and replace it by constraint (16) imposing that $(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) \in \operatorname{conv}(S)$;
- we solve this relaxation, which requires to provide a description of the convex hull of set S, by means of column generation techniques;
- as the resulting solution may violate the relaxed requirements, the column generation scheme is embedded within an enumerative algorithm which branches on first-stage variables only;
- our algorithm first branches on integer $\boldsymbol{x}$ variables having a fractional value, and then possibly resorts to spatial branching on continuous $\boldsymbol{x}$ variables, until each continuous variable attains either its lower or upper bound;
- when this is the case, according to Corollary 1, the current solution is optimal for the actual subproblem.

The resulting branch-and-price algorithm stores the best feasible solution found (the *incumbent* solution) which is returned when the method stops.

3.3.1. Node solution

Let p denote a generic node of the branching tree, associated with bounds $\boldsymbol{l}^p$ and $\boldsymbol{u}^p$ on first-stage variables.

A lower bound on the optimal solution value of node p can be computed solving the following problem

$$\begin{array}{lll} \min\limits_{\boldsymbol{x},\boldsymbol{t},\boldsymbol{y},(\boldsymbol{v}^i)_{i\in Q},\boldsymbol{\xi}} & \delta^*(\boldsymbol{\xi}|\Xi) - \sum\limits_{i\in Q}(t_i w_i)_*\left(\boldsymbol{v}^i\right) & \\ \text{subject to} & \boldsymbol{x} \in \overline{\mathcal{X}} \cap [\boldsymbol{l}^p, \boldsymbol{u}^p] & \\ & (\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) \in \operatorname{conv}(S^p) & \\ & \sum\limits_{i\in Q} \boldsymbol{v}^i = \boldsymbol{\xi} & \\ & \boldsymbol{v}^i \in \mathbb{R}^{|U|} \qquad \forall i \in Q & \\ & \boldsymbol{\xi} \in \mathbb{R}^{|U|}, & \end{array} \tag{LB^p}$$

where $S^p = \{(\boldsymbol{x}, \boldsymbol{t}, \boldsymbol{y}) : \boldsymbol{l}^p \leq \boldsymbol{x} \leq \boldsymbol{u}^p,\ x_j \in \mathbb{Z}, \forall j \in I_I,\ (\boldsymbol{t}, \boldsymbol{y}) \in \mathcal{Y}'(\boldsymbol{x})\}$. This problem is exactly the continuous relaxation of problem (P) where the bounds $\boldsymbol{l}$ and $\boldsymbol{u}$ have been replaced by $\boldsymbol{l}^p$ and $\boldsymbol{u}^p$. Note that at the root node we have $\boldsymbol{l}^0 = \boldsymbol{l}$ and $\boldsymbol{u}^0 = \boldsymbol{u}$.

Let $(\boldsymbol{x}^{p*}, \boldsymbol{t}^{p*}, \boldsymbol{y}^{p*}, (\boldsymbol{v}^{ip*})_{i \in Q}, \boldsymbol{\xi}^{p*})$ be an optimal solution of problem LB^p. If $v(\text{LB}^p)$ is greater than or equal to the cost of the incumbent, the node is fathomed by bounding. Otherwise, we distinguish three cases:

- if $\boldsymbol{x}^{p*} \in \text{vert}\,([\boldsymbol{l}^p, \boldsymbol{u}^p])$, by Proposition 2, this solution is optimal for the current node. Hence, the node is fathomed by optimality and the incumbent is updated;
- if $\boldsymbol{x}^{p*} \in \mathcal{X} \setminus \text{vert}\,([\boldsymbol{l}^p, \boldsymbol{u}^p])$, we compute a feasible solution for (2SRO-P) by solving the following model

$$\begin{aligned} \min_{\boldsymbol{t}, \boldsymbol{y}, (\boldsymbol{v}^i)_{i \in Q}, \boldsymbol{\xi}} \quad & \delta^*(\boldsymbol{\xi} | \Xi) - \sum_{i \in Q} (t_i w_i)_* \left(\boldsymbol{v}^i\right) \\ \text{subject to} \quad & (\boldsymbol{t}, \boldsymbol{y}) \in \text{conv}\,(\mathcal{Y}'(\boldsymbol{x}^{p*})) \\ & \sum_{i \in Q} \boldsymbol{v}^i = \boldsymbol{\xi} \\ & \boldsymbol{v}^i \in \mathbb{R}^{|U|} \qquad \forall i \in Q \\ & \boldsymbol{\xi} \in \mathbb{R}^{|U|}, \end{aligned} \tag{UB^p}$$

  in which the first-stage variables are fixed to $\boldsymbol{x}^{p*}$. Note that, in this case, $\boldsymbol{x}^{p*}$ is a feasible first-stage solution; hence, by Assumption 3, problem UB^p is always feasible, and possibly the incumbent is updated. If $v(LB^p) = v(UB^p)$ then node p is solved; otherwise, we perform a branching;
- if $\boldsymbol{x}^{p*} \in \overline{\mathcal{X}} \setminus \mathcal{X}$, we branch.

In the last case, before branching, one can try to round $\boldsymbol{x}^{p*}$; if the resulting point is in $\mathcal{X}$, a feasible solution for (2SRO-P) can be computed.

3.3.2. Branching

We now describe how to select the branching variable at node p. For each first-stage variable, say with index $j \in I$, we compute the minimum distance of x_j^{p*} from one of its bounds at the node, i.e., we evaluate

$$\theta_j^p = \begin{cases} \min\{x_j^{p*} - \lfloor x_j^{p*} \rfloor; \lceil x_j^{p*} \rceil - x_j^{p*}\} & \text{if } j \in I_I \\ \min\{x_j^{p*} - l_j^p; u_j^p - x_j^{p*}\} & \text{otherwise.} \end{cases}$$

(a) Left child ($x \leq \beta$) (b) Right child ($x \geq \beta$)

Figure 2: Branching on continuous variable x from example 1

For branching, we give priority to integer variables that do not attain their bound. Otherwise, we resort to spatial branching on continuous variables. In both cases, we select the variable with maximum θ_j^p value, i.e., we select variable $x_{\bar{j}}$ such that

$$\bar{j} \in \begin{cases} \text{argmax}\{\theta_j^p : j \in I_I\} & \text{if } \exists j \in I_I : \theta_j > 0 \\ \text{argmax}\{\theta_j^p : j \in I_C\} & \text{otherwise.} \end{cases}$$

If $\bar{j} \in I_I$, then a standard integer branching is executed. Otherwise, spatial branching generates two descendant nodes by imposing $x_{\bar{j}} \leq x_{\bar{j}}^{p*}$ for the left node and $x_{\bar{j}} \geq x_{\bar{j}}^{p*}$ for the right one. We associate to each node the lower bound value of the current node $v(\text{LB}^p)$ and insert them in a list of open nodes. At each iteration, we extract from the list one node with minimum lower bound value, halting the algorithm when the list is empty.

Example 5. *Figure 2 illustrates the feasible region of the left and right child obtained by spatial branching on $x \leq \beta$ and $x \geq \beta$, respectively, from example 1 (here, $\beta = 0.4$). Clearly, the right child allows the same second-stage decisions as in $\mathcal{Y}(x)$ for all $x \geq \beta$. The left child, however, still allows second-stage decisions that could end up being infeasible in the original problem. In particular, $(\boldsymbol{x}, \boldsymbol{y}) = (\varepsilon, 1-\varepsilon, 0)$ with $\varepsilon \in (0, \beta]$ is feasible for* (LB^p) *but not for* (2SRO-P).

3.3.3. Convergence

In the following, we analyze the convergence of the branch-and-price algorithm. While finite convergence is ensured if all first-stage variables are integer, this may not be the case when the first-stage includes continuous variables. We now consider the case where our algorithm has an infinite number of nodes. Note that, in this case, there exists at least one infinite

branch to the branching tree since the number of variables which can be selected for branching is finite. We consider one such branch and denote it by P. For each node $p \in P$, we denote by $(\boldsymbol{l}^p, \boldsymbol{u}^p)$ the associated bounds for the $\boldsymbol{x}$ variables and by $(\boldsymbol{x}^{p*}, \boldsymbol{t}^{p*}, \boldsymbol{y}^{p*}, \boldsymbol{V}^{p*}, \boldsymbol{\xi}^{p*})$ an optimal solution to the lower-bounding problem. Additionally, we introduce the following function

$$f_{LB}(\boldsymbol{t}, \boldsymbol{V}, \boldsymbol{\xi}) := \delta^*(\boldsymbol{\xi}|\Xi) - \sum_{i \in Q} (t_i w_i)_* \left(\boldsymbol{v}^i\right), \tag{17}$$

which gives the value of the lower bounding problem (P) as a function of its arguments. Similarly, from Lemma 2, function

$$f_{ST}(\boldsymbol{t}) := \sup_{\boldsymbol{\xi} \in \Xi} \sum_{i \in Q} w_i(\boldsymbol{\xi}) t_i. \tag{18}$$

gives the value of the single stage reformulation as a function of argument $\boldsymbol{t}$.

Remark 6. For each node $p \in P$ it holds $f_{LB}(\boldsymbol{t}^{p*}, \boldsymbol{V}^{p*}, \boldsymbol{\xi}^{p*}) = f_{ST}(\boldsymbol{t}^{p*})$.

Proof. This directly follows from the definition of $(\boldsymbol{x}^{p*}, \boldsymbol{t}^{p*}, \boldsymbol{y}^{p*}, \boldsymbol{V}^{p*}, \boldsymbol{\xi}^{p*})$ and Proposition (1). □

Lemma 4. *Let P be a sequence of nodes of any infinite branch of the branching tree. Then,*

(i) The sequence $\{(\boldsymbol{l}^p, \boldsymbol{u}^p)\}_{p \in P}$ has a unique accumulation point, which we denote by $(\boldsymbol{l}^*, \boldsymbol{u}^*)$;
(ii) The sequence $\{(\boldsymbol{x}^{p*}, \boldsymbol{t}^{p*}, \boldsymbol{y}^{p*})\}_{p \in P}$ has at least one accumulation point;
(iii) Let $\boldsymbol{x}^*$ be any accumulation point of $\{\boldsymbol{x}^{p*}\}_{p \in P}$, then, for each $j \in I_C$ which is infinitely selected for branching, there exists a sub-sequence $P^j \subseteq P$ such that either $\{u_j^p\}_{p \in P^j} \to x_j^*$ or $\{l_j^p\}_{p \in P^j} \to x_j^*$;
(iv) Every accumulation point $\boldsymbol{x}^*$ of $\{\boldsymbol{x}^{p*}\}_{p \in P}$ satisfies $\boldsymbol{x}^* \in \operatorname{vert}([\boldsymbol{l}^*, \boldsymbol{u}^*])$.

Proof.

(i) This follows from the fact that $\boldsymbol{l}^p$ (resp. $\boldsymbol{u}^p$) is a bounded, non-decreasing (resp. non-increasing) sequence.
(ii) This follows from the Bolzano-Weierstrass theorem since the sequence $\{\boldsymbol{x}^{p*}\}_{p \in P}$ is generically bounded by $[\boldsymbol{l}, \boldsymbol{u}]$, $\overline{\mathcal{X}}$ is compact and $\operatorname{conv}(S)$ is closed and bounded, thus compact (indeed, for all $i \in Q$, t_i^p is trivially bounded by $\sup\{\varphi_i(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x} \in \overline{\mathcal{X}}, \boldsymbol{y} \in \overline{\mathcal{Y}}(\boldsymbol{x})\}$ which is finite by Assumption 4.1).

(iii) Consider any accumulation point $\boldsymbol{x}^*$ of $\{\boldsymbol{x}^{p*}\}_{p\in P}$ with its associated convergent sub-sequence $P' \subseteq P$, i.e., $\{\boldsymbol{x}^{p*}\}_{p\in P'} \to \boldsymbol{x}^*$. Let $j \in I_C$ be as described in the lemma, and consider the sub-sequence $P^{u_j} \subseteq P'$ such that, for all $p \in P^{u_j}$, $u_j^{p+1} = x_j^{p*}$. Assume P^{u_j} is not finite. Then, we have that $\{x_j^{p*}\}_{p\in P^{u_j}} \to x_j^*$ since $P^{u_j} \subseteq P'$. And thus, by definition of P^{u_j}, we have that $\{u_j^{p+1}\}_{p\in P^{u_j}} \to x_j^*$. We therefore chose $P^j = \{p+1 : p \in P^{u_j}\}$ and have $\{u_j^p\}_{p\in P^j} \to x_j^*$. If instead P^{u_j} is finite, the sub-sequence $P^{l_j} \subseteq P'$ defined by nodes p for which $l_j^{p+1} = x_j^{p*}$ is infinite; therefore, the similar argument can be applied.

(iv) We have just shown that, for any accumulation point $\boldsymbol{x}^*$ of $\{\boldsymbol{x}^{p*}\}_{p\in P}$, with its associated convergent sub-sequence $P' \subseteq P$, and any infinitely branched index $j \in I_C$, there exists $P^j \subseteq P'$ such that either $\{u_j^p\}_{p\in P^j} \to x_j^*$ or $\{l_j^p\}_{p\in P^j} \to x_j^*$. Assume $\{l_j^p\}_{p\in P^j} \to x_j^*$ holds. Then, we have that $P^j \subseteq P'$ and $\{l_j^p\}_{p\in P'} \to l_j^*$. Thus, $x_j^* = l_j^*$ holds, since any sub-sequence of a converging sequence converges to the same point. The same argument can be applied when $\{u_j^p\}_{p\in P^j} \to x_j^*$.

□

Theorem 1. *Let P be a sequence of nodes of any infinite branch of the branching tree. Then, every accumulation point of $\{(\boldsymbol{x}^p, \boldsymbol{t}^p, \boldsymbol{y}^p)\}_{p\in P}$, say $(\boldsymbol{x}^*, \boldsymbol{t}^*, \boldsymbol{y}^*)$, is an optimal solution of problem* (7)*, and, thus, $\boldsymbol{x}^*$ is an optimal solution of* (2SRO-P)*.*

Proof. By Lemma 4 (ii), there exists a sub-sequence $P' \subseteq P$ such that $\{(\boldsymbol{x}^{p*}, \boldsymbol{t}^{p*}, \boldsymbol{y}^{p*})\}_{p\in P'} \to (\boldsymbol{x}^*, \boldsymbol{t}^*, \boldsymbol{y}^*)$. Note that $\overline{\mathcal{X}}$ and $\operatorname{conv}(S)$ are compact sets, hence we have that $(\boldsymbol{x}^*, \boldsymbol{t}^*, \boldsymbol{y}^*) \in \overline{\mathcal{X}} \times \operatorname{conv}(S)$. Moreover, by Lemma 4 (iv), we know that $\boldsymbol{x}^* \in \operatorname{vert}([\boldsymbol{l}^*, \boldsymbol{u}^*])$ which, by Proposition 2, ensures that $(\boldsymbol{t}^*, \boldsymbol{y}^*) \in \operatorname{conv}(\mathcal{Y}'(\boldsymbol{x}^*))$. Hence, $(\boldsymbol{x}^*, \boldsymbol{t}^*, \boldsymbol{y}^*)$ is feasible for (7). Note that f_{ST} is a continuous function since it is the point-wise supremum of continuous (affine) functions. Thus, by Remark 6, we have $\{f_{LB}(\boldsymbol{t}^{p*}, \boldsymbol{V}^{p*}, \boldsymbol{\xi}^{p*})\}_{p\in P'} \to f_{ST}(\boldsymbol{t}^*)$. In other words, the objective value of the feasible solution $(\boldsymbol{x}^*, \boldsymbol{t}^*, \boldsymbol{y}^*)$ to (7) is $f_{ST}(\boldsymbol{t}^*)$. Yet, by Lemma 2, we know that (7) and (2SRO-P) have the same objective value. This makes $\boldsymbol{x}^*$ a feasible solution to (2SRO-P) of value $f_{ST}(\boldsymbol{t}^*)$. Since our node selection strategy always picks a node with minimum lower bound, for each node $p \in P$, we have $f_{LB}(\boldsymbol{t}^{p*}, \boldsymbol{V}^{p*}, \boldsymbol{\xi}^{p*}) = v(LB^p) \leq v(\text{2SRO-P}) \leq f_{ST}(\boldsymbol{t}^*)$. As $v(LB^p)$ converges to $f_{ST}(\boldsymbol{t}^*)$, we also have $f_{LB}(\boldsymbol{t}^{p*}, \boldsymbol{V}^{p*}, \boldsymbol{\xi}^{p*}) = f_{ST}(\boldsymbol{t}^*) = v(\text{2SRO-P})$. □

We conclude this section by observing that, at each node of the branch-and-price algorithm, the lower bounding problem can be solved with ε-tolerance in a finite number of operations. Indeed, as shown in Ceria and Soares (1999) and Grossmann and Ruiz (2012), one can reformulate a convex disjunctive program as a convex MINLP by introducing an exponential number of auxiliary variables that model the disjunctions $\bigcup_{k\in\mathbb{Z}\cap[l_j,u_j]}\{x_j = k\}$ for each $j \in I_I$. The resulting model can thus be solved in finite number of steps by using any algorithm designed for convex optimization.

3.4. A convexification scheme based on column-generation

In this section, we propose a nonlinear column-generation algorithm to be used, at each node p, to solve problem (LB^p) to ε-optimality in a finite number of iterations. According to this scheme, we approximate $\mathrm{conv}\,(S^p)$ by the convex hull of a finite set of points belonging to S^p.

Restricted Master Problem: To determine this set, we use an iterative approach. At each iteration k, let $K = \{1, \ldots, k\}$ and denote by $H^{pk} = \{(\bar{\mathbf{x}}^{pj}, \bar{\mathbf{t}}^{pj}, \bar{\mathbf{y}}^{pj}) : j \in K\}$ the current set of points in the restricted master. As $H^{pk} \subseteq S^p$, we have $\mathrm{conv}\left(H^{pk}\right) \subseteq \mathrm{conv}\,(S^p)$, thus the optimal solution of the problem obtained by substituting $\mathrm{conv}\,(S^p)$ with $\mathrm{conv}\left(H^{pk}\right)$ in (LB^p) gives an upper bound of (LB^p). The resulting problem, denoted as ($\widehat{\mathrm{LB}}^{pk}$),

is called the *Restricted Master*, and is formulated as follows.

$$\min_{\boldsymbol{x},\boldsymbol{t},\boldsymbol{y},\boldsymbol{V},\boldsymbol{\xi},\boldsymbol{\alpha}} \delta^*(\boldsymbol{\xi}|\Xi) - \sum_{i\in Q}(t_i w_i)_*\left(\boldsymbol{v}^i\right) \tag{20}$$

$$\text{subject to } \boldsymbol{x} \in \overline{\mathcal{X}} \cap [\boldsymbol{l}^p, \boldsymbol{u}^p] \tag{21}$$

$$\boldsymbol{x} = \sum_{j\in K} \alpha_j \bar{\mathbf{x}}^{pj} \tag{22}$$

$$\boldsymbol{t} = \sum_{j\in K} \alpha_j \bar{\mathbf{t}}^{pj} \tag{23}$$

$$\boldsymbol{y} = \sum_{j\in K} \alpha_j \bar{\mathbf{y}}^{pj} \tag{24}$$

$$\sum_{j\in K} \alpha_j = 1 \tag{25}$$

$$\sum_{i\in Q} \boldsymbol{v}^i = \boldsymbol{\xi} \tag{26}$$

$$\boldsymbol{v}^i \in \mathbb{R}^{|U|} \qquad \forall i \in Q \tag{27}$$

$$\boldsymbol{\xi} \in \mathbb{R}^{|U|} \tag{28}$$

$$\alpha_j \geq 0 \qquad \forall j \in K. \tag{29}$$

$$(\widehat{\mathrm{LB}}^{pk})$$

Following the classical column-generation framework, the current approximation can be improved by means of a so-called *Pricing Problem*, aimed at identifying new points to be added to the Restricted Master, and defined as follows.

Pricing Problem: Let $\boldsymbol{\lambda}^{pk*}, \boldsymbol{\mu}^{pk*}, \boldsymbol{\pi}^{pk*}$ and η^{pk*} be the values of the dual variables associated with constraints (22), (23), (24), and (25) in an optimal solution of problem $(\widehat{\mathrm{LB}}^{pk})$.

Pricing asks to solve the following problem

$$(\bar{\mathbf{x}}^{p,k+1}, \bar{\mathbf{t}}^{p,k+1}, \bar{\mathbf{y}}^{p,k+1}) \in \operatorname*{argmin}_{(\boldsymbol{x},\boldsymbol{t},\boldsymbol{y})\in S^p} -\boldsymbol{\lambda}^{pk*^T}\boldsymbol{x} - \boldsymbol{\mu}^{pk^T}\boldsymbol{t} - \boldsymbol{\pi}^{pk*^T}\boldsymbol{y} - \eta^{pk*^T} \tag{PP^{pk}}$$

and generates a new point $(\bar{\mathbf{x}}^{p,k+1}, \bar{\mathbf{t}}^{p,k+1}, \bar{\mathbf{y}}^{p,k+1})$ belonging to S^p. If $v(\mathrm{PP}^{pk}) \geq -\varepsilon$, we have an ε-optimal solution to (LB^p), and hence the algorithm terminates. Otherwise, we set $H^{k+1} = H^k \cup \{(\bar{\mathbf{x}}^{p,k+1}, \bar{\mathbf{t}}^{p,k+1}, \bar{\mathbf{y}}^{p,k+1})\}$, $k = k+1$ and iterate. Note that, at each iteration k, a lower bound on the optimal

solution value of (LB^p) is given by $v(\widehat{\mathrm{LB}}^{pk}) - v(\mathrm{PP}^{pk})$. This lower bound, combined with an upper bound, can allow us to early terminate the solution of problem (LB^p).

4. Computational experiments

In this section, we report computational results of our solution algorithm when applied to an uncertain Capacitated Facility Location Problem with congestion.

4.1. Problem definition

We consider a variant of the Facility Location Problem, in which we are given a set V_1 of candidate sites for opening facilities, as well as a set V_2 of clients to be served with some product. Each client $j \in V_2$ has a demand d_j representing the quantity of product that she/he wants to receive. Each site $i \in V_1$ can be activated at a given fixed cost $f_i > 0$. In this case, one has to decide the capacity to be installed, at cost u_i per unit of capacity. Each site i has an upper bound $\bar{q}_i$ on the maximum capacity that can be installed. Each connection $(i, j) \in V_1 \times V_2$ is associated with a fixed cost c_{ij}, and a variable cost t_{ij} per unit of product which is transported. In our setting, we explicitly model congestion at each site i by means of an additional cost which depends on the total amount of product, say o_i, leaving the facility. As in the congested Facility Location Problem considered in Desrochers et al (1995) and in Fischetti et al (2016), the congestion cost for site i is given by

$$F_i(o_i) = (\alpha_i + \beta_i o_i^{\gamma_i}) o_i, \tag{29}$$

where $\alpha_i \geq 0$, $\beta_i > 0$ and $\gamma_i \geq 1$ are input parameters. Note that each function F_i is convex for non-negative arguments o_i. The problem asks to determine the facilities to be opened, the capacity to be installed at each facility and the flow of product from facilities to clients, so as to serve all clients at minimum cost. This problem can be reduced to the one addressed in Desrochers et al (1995) in case there are no capacity constraints at the sites (i.e., $\bar{q}_i = \infty$ and $u_i = 0$ for each $i \in V_1$) and transportation costs only include a variable component (i.e., $c_{ij} = 0$ for each $(i, j) \in V_1 \times V_2$).

In our context, connection costs are not known when deciding the capacities to be installed. Formally, for each connection $(i, j) \in V_1 \times V_2$, we denote by $\bar{c}_{ij}$ and $\bar{t}_{ij}$ the nominal fixed and variable costs from i to j, and by $\tilde{c}_{ij}$

and $\tilde{t}_{ij}$ their maximal deviations. Without loss of generality, we assume that, for each connection $(i,j) \in V_1 \times V_2$, the actual realizations for the costs are determined by the same variable ξ_{ij}. In other words, we have $c_{ij} = \bar{c}_{ij} + \xi_{ij}\tilde{c}_{ij}$ and $t_{ij} = \bar{t}_{ij} + \xi_{ij}\tilde{t}_{ij}$, with $\boldsymbol{\xi} \in \Xi$ and $\Xi \subseteq [0,1]^{|V_1|\times|V_2|}$ is a given uncertainty set (see Section 4.2).

We consider the adjustable robust version of this uncertain problem, where capacity installation is determined at the first stage whereas product flows are determined after uncertainty reveals. We denote the resulting problem as ARCCFLP (for Adjustable Robust Congested Capacitated Facility Location Problem).

4.2. Mathematical formulation

To model ARCCFLP, we introduce, for each site $i \in V_1$, first-stage variables x_i and q_i; the former takes the value 1 if site i is activated, whereas the latter denotes the actual capacity installed. The feasible set X for the first-stage variables is defined as

$$X = \{(\boldsymbol{x}, \boldsymbol{q}) : x_i \in \{0,1\} \text{ and } 0 \leq q_i \leq \bar{q}_i x_i \qquad \forall i \in V_1\}. \tag{30}$$

Once the actual realization of uncertainty $\boldsymbol{\xi} \in \Xi$ is known, thus defining the transportation costs, the remaining decisions concerning the flow of product from opened sites to clients must be taken. To this aim we introduce, for each connection $(i,j) \in V_1 \times V_2$, variables b_{ij} and y_{ij} denoting if the connection is activated and the fraction of request of client j that is served by site i, respectively. For each site i, we also denote by o_i the total amount of product leaving the site. Accordingly, the feasible set $Y(\boldsymbol{x}, \boldsymbol{q})$ associated with a given pair $(\boldsymbol{x}, \boldsymbol{q})$ is defined by the following constraints.

$$o_i = \sum_{j \in V_2} d_j y_{ij} \leq q_i \qquad \forall i \in V_1 \tag{31}$$

$$\sum_{i \in V_1} y_{ij} = 1 \qquad \forall j \in V_2 \tag{32}$$

$$y_{ij} \leq b_{ij} \qquad \forall i \in V_1, \forall j \in V_2 \tag{33}$$

$$y_{ij} \geq 0 \qquad \forall i \in V_1, \forall j \in V_2 \tag{34}$$

$$b_{ij} \in \{0,1\} \qquad \forall i \in V_1, \forall j \in V_2. \tag{35}$$

Constraints (31) enforce that the total demand o_i leaving each site i does not exceed the installed capacity, while constraints (32) impose that, for each

client, all the demand is served. Constraints (33) activate connections with a positive flow. Finally, (34) and (35) give the domain of the variables.

In order to ensure complete recourse, we introduce a dummy facility k with $f_k = u_k = 0$, $\bar{q}_k = \sum_{j \in V_2} d_j$, and with very large values for $\bar{c}_{kj}$ and $\bar{t}_{kj}$ for each $j \in V_2$. By adding constraints $x_k = 1$ and $q_k = \bar{q}_k$ in the definition of set X, we force facility k to be opened at maximum capacity. This choice has zero cost in the first stage, and allows a feasible solution with very large cost in the second stage regardless the values of the remaining first-stage variables.

Then, ARCCFLP is formulated as

$$\min_{(\boldsymbol{x},\boldsymbol{q}) \in X} \left\{ \sum_{i \in V_1} (f_i x_i + u_i q_i) \right.$$
$$\left. + \max_{\boldsymbol{\xi} \in \Xi} \min_{(\boldsymbol{b},\boldsymbol{y},\boldsymbol{o}) \in Y(\boldsymbol{x},\boldsymbol{q})} \sum_{i \in V_1} \left(F_i(o_i) + \sum_{j \in V_2} \left((\bar{c}_{ij} + \xi_{ij}\tilde{c}_{ij}) b_{ij} + (\bar{t}_{ij} + \xi_{ij}\tilde{t}_{ij}) d_j y_{ij} \right) \right) \right\}. \tag{36}$$

By applying the methodology introduced in this paper, the resulting lower-bounding problem is given as follows

$$\min \sum_{i \in V_1} \left(f_i x_i + u_i q_i + r_i + \sum_{j \in V_2} (\bar{c}_{ij} b_{ij} + \bar{t}_{ij} d_j y_{ij}) \right)$$
$$+ \max_{\boldsymbol{\xi} \in \Xi} \sum_{i \in V_1} \sum_{j \in V_2} \xi_{ij} (\tilde{c}_{ij} b_{ij} + \tilde{t}_{ij} d_j y_{ij}) \tag{37}$$
$$\text{subject to } (\boldsymbol{x}, \boldsymbol{q}) \in X \tag{38}$$
$$(\boldsymbol{q}, \boldsymbol{r}, \boldsymbol{o}, \boldsymbol{y}, \boldsymbol{b}) \in \operatorname{conv} \left(\left\{ (\boldsymbol{q}, \boldsymbol{r}, \boldsymbol{o}, \boldsymbol{y}, \boldsymbol{b}) : \begin{array}{ll} 0 \le q_i \le \bar{q}_i & \forall i \in V_1 \\ r_i \ge F_i(o_i) & \forall i \in V_1 \\ (31) - (35) & \end{array} \right\} \right). \tag{39}$$

The inner maximization problem can then be expressed by using Fenchel duality, and the resulting formulation depends on the uncertain set. In our experiments, we consider two widely used uncertainty sets, namely, the Γ-uncertainty set and the ellipsoidal uncertainty set.

Γ-uncertainty is a paradigm introduced in Bertsimas and Sim (2004) to model uncertain situations in which the robustness of the solution can be

controlled by an input parameter $\Gamma > 0$. By using this uncertainty set in our context, we obtain

$$\Xi_\Gamma^{\triangleright} = \left\{ \boldsymbol{\xi} \in [0,1]^{|V_1|\times|V_2|} : \sum_{i\in V_1}\sum_{j\in V_2} \xi_{ij} \le \Gamma \right\}. \tag{40}$$

In this case, Fenchel duality reduces to LP duality as follows.

$$\max_{\boldsymbol{\xi}\in\Xi_\Gamma^{\triangleright}} \sum_{i\in V_1}\sum_{j\in V_2} \xi_{ij}(\tilde{c}_{ij}b_{ij} + \tilde{t}_{ij}d_j y_{ij}) = \min \ \ \Gamma\lambda + \sum_{i\in V_1}\sum_{j\in V_2} \pi_{ij} \tag{41}$$

$$\text{subject to } \lambda + \pi_{ij} \ge \tilde{c}_{ij}b_{ij} + \tilde{t}_{ij}d_j y_{ij} \quad \forall i \in V_1, \forall j \in V_2 \tag{42}$$

$$\lambda \ge 0 \tag{43}$$

$$\pi_{ij} \ge 0 \quad \forall i \in V_1, \forall j \in V_2. \tag{44}$$

The Ellipsoidal uncertainty set is defined as

$$\Xi_\Gamma^{\circ} = \left\{ \boldsymbol{\xi} \in [0,1]^{|V_1|\times|V_2|} : \sqrt{\sum_{i\in V_1}\sum_{j\in V_2} \xi_{ij}^2} \le \Gamma \right\}, \tag{45}$$

where again Γ is a control parameter. In this case, one obtains the following formulation.

$$\max_{\boldsymbol{\xi}\in\Xi_\Gamma^{\circ}} \sum_{i\in V_1}\sum_{j\in V_2} \xi_{ij}(\tilde{c}_{ij}b_{ij} + \tilde{t}_{ij}d_j y_{ij}) = \min \ \ \Gamma\lambda + \sum_{i\in V_1}\sum_{j\in V_2} \pi_{ij} \tag{46}$$

$$\text{subject to } \nu_{ij} + \pi_{ij} \ge \tilde{c}_{ij}b_{ij} + \tilde{t}_{ij}d_j y_{ij} \quad \forall i \in V_1, \forall j \in V_2 \tag{47}$$

$$\sqrt{\sum_{i\in V_1}\sum_{j\in V_2} \nu_{ij}^2} \le \lambda \tag{48}$$

$$\lambda \ge 0 \tag{49}$$

$$\pi_{ij} \ge 0 \quad \forall i \in V_1, \forall j \in V_2 \tag{50}$$

$$\nu_{ij} \ge 0 \quad \forall i \in V_1, \forall j \in V_2. \tag{51}$$

An interested reader may refer to Li et al (2011) for associated theoretical properties of both uncertainty sets, including their robust counterparts and probabilistic guarantees for linear constraints.

4.3. Test bed

Instance generation. We tested our solution method on random instances, that were generated by following the guidelines of the extensive computational study by Cornuejols et al (1991). Accordingly, for each facility $i \in V_1$, the maximum capacity $\bar{q}_i$ and the fixed opening cost f_i follow uniform distributions in $[10, 160]$ and $[0, 180]$, respectively, whereas the variable coefficient u_i was generated in $[200/\sqrt{\bar{q}_i}, 220/\sqrt{\bar{q}_i}]$. Moreover, locations for each facility $i \in V_1$ and each client $j \in V_2$ were generated in the unit square, and nominal transportation costs were set to the Euclidean distance multiplied by 10 and rounded up. The demands were uniformly generated between 0 and 1 and scaled so that $\sum_{i \in V_1} \bar{q}_i / \sum_{j \in V_2} d_j = \nu$ where ν is a parameter taking value in $\{1.1, 1.2, 1.3\}$. In addition, following Desrochers et al (1995), for each $i \in V_1$, we used $\gamma_i = 1$ and $\alpha_i = \beta_i = 0.75$, i.e., each function F_i is quadratic with respect to the amount of product leaving site i. Concerning the parameters affected by uncertainty, the maximum deviation $\tilde{t}_{ij}$ was set to $0.50 \times \bar{t}_{ij}$, thus allowing a maximum of 50% deviation. As to the opening cost of each arc, we randomly generated the nominal value between 50 and 100, allowing a maximum of 50% deviation with respect to this value.

Finally, the number of sites and clients take values $(4, 8)$, $(5, 10)$ and $(6, 12)$. For each combination of $|V_1|$, $|V_2|$, and ν, we generated 5 test-cases. Each test-case was solved for $\Gamma = 1, 2, 3, 4$, both in the Γ-uncertainty and in the Ellipsoidal uncertainty settings, thus producing 360 instances.

4.4. Implementation details

We implemented our branch-and-price algorithm with spatial branching in C++ and run all the experiments on an AMD Ryzen 5 PRO 4650GE at 3.3 GHz, with a time limit equal to 10,800 CPU seconds per run (3 hours).

At the root node, an initial upper bound is computed by solving the single-stage version of ARCCFLP where all decisions are taken here and now. This bound is obtained by solving (37)-(39) without convexifying the second-stage feasible region in constraint (39). At each node of the algorithm, we solve the restricted master problem by using Mosek 10.0.36, and the pricing problem by means of Gurobi 10.0. This combination of solvers turned out to be the most numerically stable on our instances. Our code and instances are publicly available on GitHub[1].

[1] `https://github.com/hlefebvr/AB_AdjustableRobustOptimizationWithObjectiveUncertainty`

The column-generation procedure includes stabilization by dual price smoothing, as described in Pessoa et al (2013); and at most one column is added to the restricted master problem at each iteration.

For the branching strategy of continuous variables, we used a tolerance of $\varepsilon = 10^{-3}$ for comparing real numbers in finite precision. At each node, local bounds derived from branching decisions are applied to the column generation sub-problem. The restricted master inherits all columns from the father node. For each column, we compare the value of the first-stage variables in the colum with the actual bound at the node, and possibly remove it from the master when locally infeasible.

At a given node, to check the optimality of a first-stage decision and possibly fathom the node, we use the sufficient condition from Proposition 2. If the latter does not hold, we check if all active columns at optimality are built on the same values for the continuous variables; in this case, Proposition 2 can be exploited to ensure local optimality of the solution. If the first-stage solution is not feasible, an upper bound is computed as follows. We detect the variable with largest value in the RMP current solution, recover the values of variables $(\boldsymbol{x}, \boldsymbol{q})$ that were used for generating this column and fix variables $(\boldsymbol{x}, \boldsymbol{q})$ to those values, possibly rounding up integer variables.

Finally, observe that branching may induce infeasibility in the second stage. To early detect this situation, we consider a problem-specific improvement: at a given node of the branching tree, we check if $\sum_{i \in V_1} x_i^u q_i^u < \sum_{j \in V_2} d_j$ holds, where x_i^u and q_i^u denote the local upper bound of variable x_i and q_i, respectively. In this case the node is declared infeasible.

4.5. General results

Table 1 reports our computational results on ARCCFLP. The upper part relates to experiments done with the Γ-uncertainty set (Γ-unc.), while the lower part addresses those with the ellipsoidal uncertainty set (Ellips.). Columns $|V_1|$, $|V_2|$ and Γ give the number of sites, the number of clients and the value for the uncertainty parameter Γ, respectively. Column "solved" reports the number of instances (out of 15) which could be solved to proven optimality within the given time limit. Into brackets we report the number of instances for which the computation was stopped due to numerical issues of the used solvers. For the sake of consistency, all remaining columns but the last one are relative to instances which could be solved within the time limit. In particular, columns "time" report, from left to right, the average time needed to prove optimality ("total"), the average time spent solving

the RMP ("RMP") and the average time spent solving the pricing problem ("PP") during the execution of our branch-and-price algorithm. All times are expressed in seconds. Column "nodes" reports the average number of explored nodes, while "columns" gives the average number of generated columns throughout the entire execution of our algorithm. Finally, we report the average gap of the root node lower bound with respect to the optimal solution value ("root") and the average optimality gap at time limit ("end") (or when the algorithm was stopped for numerical troubles). Optimality gap is computed only over those instances which could not be solved to optimality.

					Time (s)					Gap (%)	
	$\lvert V_1\rvert$	$\lvert V_2\rvert$	Γ	solved	total	RMP	PP	nodes	columns	root	end
Γ-unc.	4	8	1	15	2.45	0.12	2.04	4.73	42.60	0.21	
			2	15	3.19	0.14	2.68	5.00	56.53	0.21	
			3	15	4.75	0.26	4.03	5.27	84.60	0.22	
			4	15	4.74	0.22	4.10	5.27	78.60	0.23	
	5	10	1	15	44.83	0.78	42.35	5.80	110.00	0.29	
			2	15	21.83	0.47	20.27	6.60	82.20	0.31	
			3	15	36.59	0.64	34.70	7.53	125.87	0.33	
			4	15	38.33	0.80	36.16	9.40	166.80	0.34	
	6	12	1	11	319.17	1.92	313.31	4.82	139.55	0.17	0.24
			2	13	572.80	1.64	562.48	5.46	102.69	0.19	0.30
			3	13	1117.35	2.35	1110.73	6.23	162.92	0.22	0.23
			4	11	1261.83	4.63	1249.46	6.45	239.00	0.22	0.26
Ellips.	4	8	1	13 (2)	6.64	0.35	5.70	5.00	67.92	0.22	0.11
			2	14 (1)	7.46	0.52	6.25	5.00	88.79	0.22	0.30
			3	14 (1)	12.47	0.67	10.90	5.29	121.07	0.20	0.46
			4	13 (2)	11.16	0.40	10.08	4.54	86.77	0.21	0.09
	5	10	1	12 (3)	26.96	1.27	24.59	6.00	117.58	0.26	0.44
			2	12 (3)	83.59	2.34	79.28	7.17	206.75	0.32	0.38
			3	15	82.47	3.19	76.72	8.73	278.80	0.34	
			4	15	151.89	2.17	147.62	6.33	195.27	0.31	
	6	12	1	14 (1)	324.66	3.45	315.11	5.29	169.57	0.18	0.21
			2	14 (1)	420.62	4.01	412.94	5.86	202.21	0.22	0.21
			3	13	859.62	5.66	847.91	7.00	280.08	0.24	0.10
			4	13 (2)	1574.26	10.83	1551.17	10.69	526.38	0.22	0.15

Table 1: Computational experiments on ARCCFLP instances. Each row refers to 15 instances.

The table shows that our method is able to solve a large fraction of the instances, namely 168 in the Γ-uncertainty setting and 162 in the Ellipsoidal uncertainty setting. In most cases, the solution time required by the algorithm is quite small and solving the RMP is very fast in practice (below 7% of the total time, on average). Indeed, the most challenging subproblem solved is the pricing problem; when increasing the size of the instances the number of columns that are needed increases and each execution of the pricing problem is more time consuming. However, the solved relaxation allows to compute a very tight approximation, as the gap between lower and upper bounds at the root that is always below 0.35%, and producing small enumeration trees, in which the number of generated nodes is always below 11. Moreover, the performance of the algorithm is satisfactory also for the instances that were not solved to proven optimality, as the average residual gap at the end of the enumeration is always quite small (below 0.5%). Finally, observe that numerical issues arise only when uncertainty belongs to the Ellipsoidal uncertainty set, a nonlinear setting in which Mosek may encounter numerical instability on some instances.

4.5.1. Linearized costs

As an alternative approach for both uncertainty sets described in Section 4.2, we considered solving a linearized approximation of ARCCFLP obtained by replacing each function F_i ($i \in V_1$) by a piecewise linear approximation. By introducing L approximation points $\{\bar{o}_{il}\}_{l=1,\dots,L}$, function F_i is underestimated by the following one:

$$\tilde{F}_i(o_i) = \max_{l=1,\dots,L} \left\{ F_i(\bar{o}_{il}) + F_i'(\bar{o}_{il})(o_i - \bar{o}_{il}) \right\}. \tag{52}$$

In our experiments, we chose $L = 10$ and, for all $i \in V_1$, defined the approximation points to be equally distributed in the interval $[0, \bar{q}_i]$, i.e., $\bar{o}_{il} = \bar{q}_i(l-1)/(L-1)$ for $l = 1, \dots, L$.

Table 2 reports the results obtained by using the linearized approach. For each combination of $|V_1|$ and $|V_2|$, we give for both the exact and the linearized approaches, the number of instances solved to optimality. Moreover, for the latter we report the average and maximum percentage error introduced by the linearization, computed as $\frac{z^* - z^L}{z^*}$, where z^* and z^L denote the optimal values of an ARCCFLP instance and of its linearized counterpart, respectively. These figures are computed with respect to instances solved by both approaches only.

			exact	linearized		
	$\lvert V_1 \rvert$	$\lvert V_2 \rvert$	solved	solved	avg. err.	max. err.
Γ-unc.	4	8	60	60	0.37	0.61
	5	10	60	60	0.38	0.63
	6	12	48	42	0.39	0.56
Ellips.	4	8	54	58	0.36	0.59
	5	10	54	59	0.39	0.58
	6	12	54	59	0.40	0.56

Table 2: Comparison of exact and linearized approaches

The table shows that the linearized approach turns out to be computationally harder in the Γ-uncertainty setting, while it gives some improvement when the Ellipsoidal setting is considered; this is mainly due to the reduced number of instances for which we encountered numerical troubles. However, in both settings, linearization introduces a nonneglibile error when underestimating the true cost of a solution. The average percentage error, over all instances, is 0.38% and can be as large as 0.63%.

5. Conclusion

In this work, we studied optimization problems where part of the cost parameters are not known at decision time, and the decision flow is modeled as a two-stage process. In particular, we addressed general problems in which all constraints (including those linking the first and the second stages) are defined by convex functions and involve mixed-integer variables. To the best of our knowledge, this is the first attempt to extend the existing literature to tackle this wide class of problems.

To this aim, we derive a relaxation of the problem which can be formulated as a convex optimization problem, and embed it within an enumerative algorithm where branching occurs on integer and continuous variables. By combining enumeration and on-the-fly generation of the variables, we obtain a branch-and-price scheme, for which we prove asymptotic convergence in the general mixed-integer case and give sufficient conditions for finite convergence.

In addition to the theoretical analysis, we applied our method to an optimization problem affected by objective uncertainty arising in the logistic

field, namely a variant of the congested Capacitated Facility Location problem with uncertain transportation costs. Our computational experiments showed that the proposed method is able to solve non-trivial instances for this problems. In addition, we provide a comparison with a natural approach based on linearization of the congestion function, showing that this alternative solution method would give marginal improvements in terms of performances though introducing a nonneglibile error in terms of cost of the provided solution.

Acknowledgements

The authors are grateful to three anonymous reviewers for their constructive comments and remarks.

Funding

This research was supported by "Mixed-Integer Non Linear Optimisation: Algorithms and Application" consortium, which has received funding from the European Union's EU Framework Programme for Research and Innovation Horizon 2020 under the Marie Skłodowska-Curie Actions Grant Agreement No 764759.

References

Arslan AN, Detienne B (2022) Decomposition-based approaches for a class of two-stage robust binary optimization problems. INFORMS Journal on Computing 34(2):857–871.

Ayoub J, Poss M (2016) Decomposition for adjustable robust linear optimization subject to uncertainty polytope. Computational Management Science 13: 219–239.

Ben-Tal A, Goryashko A, Guslitzer E, et al (2004) Adjustable robust solutions of uncertain linear programs. Mathematical Programming 99(2):351–376.

Ben-Tal A, Ghaoui LE, Nemirovski A (2009) Robust Optimization. Princeton University Press.

Bertsimas D, Sim M (2004) The price of robustness. Operations Research 52(1):35–53.

Bertsimas D, Brown D, Caramanis C (2010) Theory and applications of robust optimization. SIAM Review 53: 464–501.

Bertsimas D, Caramanis C (2010) Finite adaptability in multistage linear optimization. IEEE Transactions on Automatic Control 55(12):2751–2766.

Bertsimas D, Litvinov E, Sun X, et al (2013) Adaptive robust optimization for the security constrained unit commitment problem. IEEE Transactions on Power Systems 28:52–63.

Ceria S, Soares J (1999) Convex programming for disjunctive convex optimization. Mathematical Programming 86:595–614.

Cornuejols G, Sridharan R, Thizy J (1991) A comparison of heuristics and relaxations for the capacitated plant location problem. European Journal of Operational Research 50(3):280–297.

Desrochers M, Marcotte P, Stan M (1995) The congested facility location problem. Location Science 3(1):9–23

Fischetti M, Ljubić I, Sinnl M (2016) Benders decomposition without separability: A computational study for capacitated facility location problems. European Journal of Operational Research 253(3):557–569.

Gabrel V, Lacroix M, Murat C, Remli N (2011) Robust location transportation problems under uncertain demands. Discrete Applied Mathematics 164: 100–111.

Grossmann IE, Ruiz JP (2012) Generalized disjunctive programming: A framework for formulation and alternative algorithms for MINLP optimization. In: Lee J, Leyffer S (Eds) Mixed Integer Nonlinear Programming. Springer, New York, NY, pp 93–115.

Hanasusanto GA, Kuhn D, Wiesemann W (2015) K-adaptability in two-stage robust binary programming. Operations Research 63(4):877–891.

Hassene A, Bazgan C, Vanderpooten D (2009) Min-max and min-max regret versions of combinatorial optimization problems: A survey. European Journal of Operational Research 197:427–438.

Jiang R, Zhang M, Li G, Guan Y (2014) Two-stage network constrained robust unit commitment problem. European Journal of Operational Research 234:751–762.

Kämmerling N, Kurtz J (2020) Oracle-based algorithms for binary two-stage robust optimization. Computational Optimization and Applications 77(2):539–569.

Leyffer S, Menickelly M, Munson T, Vanaret C, Wild SM (2020) A survey of nonlinear robust optimization. INFOR: Information Systems and Operational Research 58(2):342–373.

Li C, Grossmann IE (2019) A finite ε-convergence algorithm for two-stage stochastic convex nonlinear programs with mixed-binary first and second-stage variables. Journal of Global Optimization 75(4):921–947.

Li Z, Ding R, Floudas CA (2011) A comparative theoretical and computational study on robust counterpart optimization: I. Robust Linear Optimization and Robust Mixed Integer Linear Optimization. Industrial & Engineering Chemistry Research 50(18):10,567–10,603.

Perchet V, Vigeral G (2015) A minmax theorem for concave-convex mappings with no regularity assumptions. Journal of Convex Analysis 22:537–540.

Pessoa A, Sadykov R, Uchoa E, and Vanderbeck F (2013) In-out separation and column generation stabilization by dual price smoothing. SEA 2013, Lecture Notes in Computes Science vol 7933, pages 354–365. Springer Berlin Heidelberg.

Rockafellar R (1970) Convex Analysis. Princeton University Press.

Sherali HD, Fraticelli B (2002) A modification of Benders' decomposition algorithm for discrete subproblems: An approach for stochastic programs with integer recourse. Journal of Global Optimization 22:319–342.

Sherali HD, Zhu X (2006) On solving discrete two-stage stochastic programs having mixed-integer first- and second-stage variables. Mathematical Programming 108(2):597–616.

Soyster AL (1973) Technical note–convex programming with set-inclusive constraints and applications to inexact linear programming. Operations Research 21(5):1154–1157.

Subramanyam A, Gounaris CE, Wiesemann W (2019) K-adaptability in two-stage mixed-integer robust optimization. Mathematical Programming Computation 12(2):193–224.

Terry T, Epelman M, Thiele A (2009) Robust linear optimization with recourse. Technical Report, Lehigh University

Yanıkoğlu İ, Gorissen BL, den Hertog D (2019) A survey of adjustable robust optimization. European Journal of Operational Research 277(3):799–813.

Zeng B, Zhao L (2013) Solving two-stage robust optimization problems by a column-and-constraint generation method. Operations Research Letters 41:457–461.

Appendix A. Recalls of convex and concave conjugate

In this appendix we review some basic results on conjugate functions and Fenchel duality. For a detailed treatment we refer to Rockafellar (1970).

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a given function, its convex conjugate is denoted by $f^* : \mathbb{R}^n \to \mathbb{R}$ and is given by

$$f^*(\boldsymbol{\pi}) = \sup_{\boldsymbol{x} \in \mathrm{dom}(f)} \left\{ \boldsymbol{\pi}^T \boldsymbol{x} - f(\boldsymbol{x}) \right\}.$$

Similarly, we denote by g_* the concave conjugate of a given function $g : \mathbb{R}^n \to \mathbb{R}$, given by

$$g_*(\boldsymbol{\pi}) = \inf_{\boldsymbol{x} \in \mathrm{dom}(g)} \left\{ \boldsymbol{\pi}^T \boldsymbol{x} - \boldsymbol{g}(\boldsymbol{x}) \right\}.$$

Note that, if f is a proper convex function and g a proper concave function, we have that $f^{**} = f$ and $g_{**} = g$. We now state the following Fenchel duality theorem.

Theorem 2. *Let $f : \mathbb{R}^n \to \mathbb{R}$ be a proper convex function and $g : \mathbb{R}^n \to \mathbb{R}$ be a proper concave function, then*

$$\inf_{\boldsymbol{x} \in \mathrm{dom}(f) \cap \mathrm{dom}(g)} \{ f(\boldsymbol{x}) - g(\boldsymbol{x}) \} = \sup_{\boldsymbol{\pi} \in \mathrm{dom}(f^*) \cap \mathrm{dom}(g_*)} \{ g_*(\boldsymbol{\pi}) - f^*(\boldsymbol{\pi}) \}$$

or equivalently,

$$\sup_{\boldsymbol{x} \in \mathrm{dom}(f) \cap \mathrm{dom}(g)} \{ g(\boldsymbol{x}) - f(\boldsymbol{x}) \} = \inf_{\boldsymbol{\pi} \in \mathrm{dom}(g_*) \cap \mathrm{dom}(f^*)} \{ f^*(\boldsymbol{\pi}) - g_*(\boldsymbol{\pi}) \}.$$

Corollary 3 (Maximizing a concave function over a convex set)**.** *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a non-empty convex set, $g : \mathbb{R}^n \to \mathbb{R}$ be a proper concave function, then*

$$\sup_{\boldsymbol{x} \in \mathcal{X}} g(\boldsymbol{x}) = \inf_{\boldsymbol{\pi}} \left\{ \delta^*(\boldsymbol{\pi}|\mathcal{X}) - g_*(\boldsymbol{\pi}) \right\},$$

where $\delta(\boldsymbol{x}|\mathcal{X}) = \begin{cases} 0 & \text{if } \boldsymbol{x} \in \mathcal{X} \\ +\infty & \text{otherwise.} \end{cases}$

Proof. The result holds from the fact that $\sup\{g(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{X}\} = \sup\{g(\boldsymbol{x}) - \delta(\boldsymbol{x}|\mathcal{X})\}$ and by application of Fenchel duality. More precisely, $\delta(\boldsymbol{x}|\mathcal{X})$ is convex and, by non-emptiness of $\mathcal{X}$, is proper. □

Notice that Fenchel duality allows the reformulation of an optimization problem which consists in maximizing a concave function over a convex set as an unconstrained convex problem since $\delta^*(\cdot|\mathcal{X})$ and $(-g_*)(\cdot)$ are convex functions and positively weighted sums preserve convexity.

Proposition 3. *Let f be a convex function, we have $(-f)_*(\boldsymbol{\pi}) = -f^*(-\boldsymbol{\pi})$.*

Proof.

$$(-f)_*(\boldsymbol{\pi}) = \inf_{\boldsymbol{x}} \left\{ \boldsymbol{\pi}^T \boldsymbol{x} - (-f)(\boldsymbol{x}) \right\} = -\sup_{\boldsymbol{x}} \left\{ -\boldsymbol{\pi}^T \boldsymbol{x} - f(\boldsymbol{x}) \right\} = -f^*(-\boldsymbol{\pi}).$$

□

Table A.3 reports some calculus rules regarding convex conjugates. The extension to concave conjugates is straightforward.

$h(\boldsymbol{x})$	$h^*(\boldsymbol{\pi})$
Separable sums	
$h(\boldsymbol{x}^1, \boldsymbol{x}^2) = f_1(\boldsymbol{x}^1) + f_2(\boldsymbol{x}^2)$	$h^*(\boldsymbol{\pi}^1, \boldsymbol{\pi}^2) = f_1^*(\boldsymbol{\pi}^1) + f_2^*(\boldsymbol{\pi}^2)$
Scalar multiplications ($\alpha > 0$)	
$h(\boldsymbol{x}) = \alpha f(\boldsymbol{x})$	$h^*(\boldsymbol{\pi}) = \alpha f^*(\boldsymbol{\pi}/\alpha)$
Affine mapping composition ($\det \boldsymbol{A} \neq 0$)	
$h(\boldsymbol{x}) = f(\boldsymbol{A}\boldsymbol{x} + \boldsymbol{b})$	$h^*(\boldsymbol{\pi}) = f^*(\boldsymbol{A}^{-T}\boldsymbol{\pi}) - \boldsymbol{b}^T\boldsymbol{A}^{-T}\boldsymbol{\pi}$
Sum with affine functions	
$h(\boldsymbol{x}) = f(\boldsymbol{x}) + \boldsymbol{a}^T\boldsymbol{x} + \boldsymbol{b}$	$h^*(\boldsymbol{\pi}) = f^*(\boldsymbol{\pi} - \boldsymbol{a}) - \boldsymbol{b}$
Sum of functions	
$h(\boldsymbol{x}) = \sum_{i=1}^{m} f_i(\boldsymbol{x})$	$h^*(\boldsymbol{\pi}) = \inf_{\boldsymbol{v}^i, i=1,\ldots,m} \left\{ \sum_{i=1}^{m} f_i^*(\boldsymbol{v}^i) \middle\vert \sum_{i=1}^{m} \boldsymbol{v}^i = \boldsymbol{\pi} \right\}$

Table A.3: Some convex conjugate calculus rules