

On the Finite Optimal Convergence of Logic-Based Benders' Decomposition in Solving 0–1 Min-Max Regret Optimization Problems with Interval Costs

Lucas Assunção¹(✉), Andréa Cynthia Santos², Thiago F. Noronha³,
and Rafael Andrade⁴

¹ Departamento de Engenharia de Produção, Universidade Federal de Minas Gerais,
Avenida Antônio Carlos, 6627, Belo Horizonte, MG 31270-901, Brazil

lucas-assuncao@ufmg.br

² ICD-LOSI, UMR CNRS 6281, Université de Technologie de Troyes,
12, rue Marie Curie, CS 42060, 10004 Troyes Cedex, France

andrea.duhamel@utt.fr

³ Departamento de Ciência da Computação, Universidade Federal de Minas Gerais,
Avenida Antônio Carlos, 6627, Belo Horizonte, MG 31270-901, Brazil

tfn@dcc.ufmg.br

⁴ Departamento de Estatística e Matemática Aplicada, Universidade Federal
do Ceará, Campus do Pici, BL 910, Fortaleza, CE 60455-900, Brazil

rca@lia.ufc.br

Abstract. This paper addresses a class of problems under interval data uncertainty composed of *min-max regret* versions of classical 0–1 optimization problems with interval costs. We refer to them as *interval 0–1 min-max regret* problems. The state-of-the-art exact algorithms for this class of problems work by solving a corresponding mixed integer linear programming formulation in a Benders' decomposition fashion. Each of the possibly exponentially many Benders' cuts is separated on the fly through the resolution of an instance of the classical 0–1 optimization problem counterpart. Since these separation subproblems may be NP-hard, not all of them can be modeled by means of linear programming, unless $P = NP$. In these cases, the convergence of the aforementioned algorithms are not guaranteed in a straightforward manner. In fact, to the best of our knowledge, their finite convergence has not been explicitly proved for any interval 0–1 min-max regret problem. In this work, we formally describe these algorithms through the definition of a logic-based Benders' decomposition framework and prove their convergence to an optimal solution in a finite number of iterations. As this framework is applicable to any interval 0–1 min-max regret problem, its finite optimal convergence also holds in the cases where the separation subproblems are NP-hard.

L. Assunção—Partially supported by the Coordination for the Improvement of Higher Education Personnel, Brazil (CAPES).

L. Assunção—The author thanks Vitor A. A. Souza and Phillippe Samer for the valuable discussions throughout the conception of this work.

1 Introduction

Robust Optimization (RO) [14] has drawn particular attention as an alternative to stochastic programming [22] in modeling uncertainty. In RO, instead of considering a probabilistic description known *a priori*, the variability of the data is represented by deterministic values in the context of *scenarios*. A *scenario* corresponds to a parameters assignment, *i.e.*, a value is fixed for each parameter subject to uncertainty. Two main approaches are adopted to model RO problems: the *discrete scenarios model* and the *interval data model*. In the former, a discrete set of possible scenarios is considered. In the latter, the uncertainty referred to a parameter is represented by a continuous interval of possible values. Differently from the discrete scenarios model, the infinite many possible scenarios that arise in the interval data model are not explicitly given. Nevertheless, in both models, a classical (*i.e.*, parameters known in advance) optimization problem takes place whenever a scenario is established.

The most commonly adopted RO criteria are the *absolute robustness* criterion, the *min-max regret* (also known as *robust deviation* criterion) and the *min-max relative regret* (also known as *relative robustness* criterion). The absolute robustness criterion is based on the anticipation of the worst possible conditions. Solutions for RO problems under such criterion tend to be conservative, as they optimize only a worst-case scenario. On the other hand, the min-max regret and the min-max relative regret are less conservative criteria and, for this reason, they have been addressed in several works (*e.g.*, [3, 5, 17, 18, 20]). Intuitively speaking, the *regret (robust deviation)* of a solution in a given scenario is the cost difference between such solution and an optimal one for this scenario. In turn, the *relative regret* of a solution in a given scenario consists of the corresponding regret normalized by the cost of an optimal solution for the scenario considered. The *(relative) robustness* cost of a solution is defined as its maximum *(relative) regret* over all scenarios. In this sense, the min-max (relative) regret criterion aims at finding a solution that has the minimum (relative) robustness cost. Such solution is referred to as a *robust solution*.

RO versions of several combinatorial optimization problems have been studied in the literature, addressing, for example, uncertainties over costs. Handling uncertain costs brings an extra level of difficulty, such that even polynomially solvable problems become NP-hard in their corresponding robust versions [13, 19, 20, 23]. In this study, we are interested in a particular class of RO problems, namely *interval 0–1 min-max regret* problems, which consist of min-max regret versions of Binary Integer Linear Programming (BILP) problems with interval costs. Notice that a large variety of classical optimization problems can be modeled as BILP problems, including (i) polynomially solvable problems, such as the shortest path problem, the minimum spanning tree problem and the assignment problem, and (ii) NP-hard combinatorial problems, such as the 0–1 knapsack problem, the set covering problem, the traveling salesman problem and the restricted shortest path problem [9]. An especially challenging subclass of interval 0–1 min-max regret problems, referred to as *interval 0–1 robust-hard*

problems, arises when we address interval 0–1 min-max regret versions of classical NP-hard combinatorial problems as those aforementioned in (ii).

Aissi et al. [1] showed that, for any interval 0–1 min-max regret problem (including interval 0–1 robust-hard problems), the robustness cost of a solution can be computed by solving a single instance of the classical optimization problem counterpart (*i.e.*, costs known in advance) in a particular scenario. Therefore, one does not have to consider all the infinite many possible scenarios during the search for a robust solution, but only a subset of them, one for each feasible solution. Nevertheless, since the number of these promising scenarios can still be huge, the state-of-the-art exact algorithms for interval 0–1 min-max regret problems work by implicitly separating them on the fly, in a Benders' decomposition [4] fashion (see, *e.g.*, [17, 18, 21]). Precisely, each Benders' cut is generated through the resolution of an instance of the classical optimization problem counterpart. Notice that, for interval 0–1 robust-hard problems, these separation subproblems are NP-hard and, thus, they cannot be modeled by means of Linear Programming (LP), unless $P = NP$. In these cases, the convergence of the aforementioned algorithms are not guaranteed in a straightforward manner.

These exact algorithms, which have their roots in *logic-based Benders' decomposition* [11] (also see [6]), have been successfully applied to solve several interval 0–1 min-max regret problems (*e.g.*, [17, 19, 20]), including interval 0–1 robust-hard problems, such as the robust traveling salesman problem [18] and the robust set covering problem [21]. However, to the best of our knowledge, their convergence has not been explicitly proved for any interval 0–1 min-max regret problem. In this work, we formally describe these algorithms through the definition of a logic-based Benders' decomposition framework and prove their finite convergence to an optimal solution. Precisely, we show, by contradiction, that a new cut is always generated per iteration, in a finite space of possible solutions. As the framework is applicable to any interval 0–1 min-max regret problem, its finite optimal convergence also holds in solving interval 0–1 robust-hard problems, *i.e.*, in the cases where the separation subproblems are NP-hard.

The remainder of this work is organized as follows. The Benders' decomposition method is briefly introduced in Sect. 2, followed by the description of a standard modeling technique for interval 0–1 min-max regret problems (Sect. 2.1). In addition, a generalization of state-of-the-art exact algorithms for interval 0–1 min-max regret problems is devised through the description of a logic-based Benders' decomposition framework (Sect. 2.2). The finite convergence of these algorithms to optimal solutions is proved in the same section, and concluding remarks are given in the last section.

2 A Logic-Based Benders' Decomposition Framework for Interval 0–1 Min-Max Regret Problems

The *classical Benders' decomposition method* [4] was originally proposed to tackle Mixed Integer Linear Programming (MILP) problems of the form $P : \min\{cx + dy : Ax + By \geq b, x \in \mathbb{Z}_+^{n_1}, y \geq \mathbf{0}\}$. In this case, there are n_1

integer variables and n_2 continuous ones, which are represented by the column vectors x and y , respectively, and their corresponding cost values are given by the row vectors c and d . Moreover, b is an m -dimensional column vector, and A and B are $m \times n_1$ and $m \times n_2$ matrices, respectively. Given a vector $\bar{x} \in \mathbb{Z}_+^{n_1}$, the *classical Benders' reformulation* starts by defining an LP *primal subproblem* $PS(\bar{x}) : \min\{dy : By \geq b - A\bar{x}, y \geq \mathbf{0}\}$ through the projection of the continuous variables y in the space defined by \bar{x} . Notice that $PS(\bar{x})$ can be represented by means of the corresponding *dual subproblem* $DS(\bar{x}) : \max\{\mu(b - A\bar{x}) : \mu B \leq d, \mu \geq \mathbf{0}\}$, where μ is an m -dimensional row vector referred to the dual variables.

Let $EP(\bar{x})$ and $ER(\bar{x})$ be, respectively, the sets of extreme points and extreme rays of a given $DS(\bar{x})$ subproblem. One may observe that the feasible region of $DS(\bar{x})$ does not depend on the value assumed by \bar{x} . Thus, hereafter, these sets are referred to as EP and ER for all $\bar{x} \in \mathbb{Z}_+^{n_1}$.

Considering a nonnegative continuous variable ρ , the resolution of each dual subproblem $DS(\bar{x})$ leads to a new linear constraint (i) $\rho \geq \bar{\mu}(b - Ax)$, if $DS(\bar{x})$ has a bounded optimal solution $\bar{\mu} \in EP$, or (ii) $\bar{\nu}(b - Ax) \leq 0$, if DP has an unbounded solution, represented by $\bar{\nu} \in ER$. The Benders' cuts described in (i) are called *optimality cuts*, whereas the ones in (ii) are called *feasibility cuts*. Both types of cuts are used to populate on the fly a *reformulated problem*, defined as:

$$(RP) \quad \min \{cx + \rho\} \tag{1}$$

$$s.t. \quad \rho \geq \bar{\mu}(b - Ax) \quad \forall \bar{\mu} \in EP, \tag{2}$$

$$\bar{\nu}(b - Ax) \leq 0 \quad \forall \bar{\nu} \in ER, \tag{3}$$

$$\rho \geq 0, \tag{4}$$

$$x \in \mathbb{Z}_+^{n_1}. \tag{5}$$

Let MRP be a relaxed RP problem, called *master problem*, which considers only a subset of the extreme points and extreme rays associated with constraints (2) and (3), respectively. At each iteration of the *classical Benders' decomposition algorithm*, the master problem is solved, obtaining a solution $(\bar{x}, \bar{\rho}) \in \mathbb{Z}_+^{n_1} \times \mathbb{R}_+$. If $(\bar{x}, \bar{\rho})$ is not feasible for the original reformulated problem RP , a corresponding $DS(\bar{x})$ subproblem is solved in order to generate a new Benders' cut, either a feasibility cut or an optimality one. The new cut is added to the master problem MRP and the algorithm iterates until a feasible (and, therefore, optimal) solution for RP is found.

The finite convergence of the classical Benders' decomposition method is guaranteed by the fact that the polyhedron referred to any LP problem can be described by finite sets of extreme points and extreme rays, and that a new Benders' cut is generated per iteration of the algorithm. We refer to [4] for the detailed proof. Methodologies able to improve the convergence of the method were studied in several works (see, e.g., [8, 15, 16]). In addition, nonlinear convex duality theory was later applied to devise a generalized approach, namely *generalized Benders' decomposition*, applicable to mixed integer nonlinear problems [10].

More recently, Hooker and Ottosson [11] introduced the idea of the so-called *logic-based Benders' decomposition*, a Benders-like decomposition approach that is suitable for a broader class of problems. In fact, the latter approach is intended to tackle any optimization problem by exploring the concept of an *inference dual subproblem*. In particular, that is the problem of inferring a strongest possible bound for a set of constraints from the original problem that are relaxed in the master problem. Notice that the aforementioned inference subproblems are not restricted to linear and nonlinear continuous problems. In fact, they can even be NP-hard combinatorial problems (see, *e.g.*, [7]). Therefore, the convergence of the logic-based Benders' decomposition method cannot be showed in a straightforward manner for all classes of optimization problems. As pointed out in [11], the convergence of the method relies on some peculiarities of the (logic-based) Benders' reformulation, such as the way the inference dual subproblems are devised and the finiteness of the search space referred to them.

In the remainder of this section, we describe a framework that generalizes state-of-the-art logic-based Benders' decomposition algorithms widely used to solve interval 0–1 min-max regret problems [17–21]. In addition, we show its convergence to an optimal solution in a finite number of iterations. The framework addresses MILP formulations with typically an exponential number of constraints, as detailed in the sequel.

2.1 Mathematical Formulation

Consider \mathcal{G} , a generic BILP minimization problem defined as follows.

$$(\mathcal{G}) \quad \min \quad cx \tag{6}$$

$$s.t. \quad Ax \geq b, \tag{7}$$

$$x \in \{0, 1\}^n. \tag{8}$$

The binary variables are represented by an n -dimensional column vector x , whereas their corresponding cost values are given by an n -dimensional row vector c . Moreover, b is an m -dimensional column vector, and A is an $m \times n$ matrix. The feasible region of \mathcal{G} is given by $\Omega = \{x : Ax \geq b, x \in \{0, 1\}^n\}$. We highlight that, although the results of this work are presented by the assumption of \mathcal{G} being a minimization problem, they also hold for interval 0–1 min-max regret versions of maximization problems, with minor modifications.

Now, let \mathcal{R} be an interval min-max regret RO version of \mathcal{G} , where a continuous cost interval $[l_i, u_i]$, with $l_i, u_i \in \mathbb{Z}_+$ and $l_i \leq u_i$, is associated with each binary variable x_i , $i = 1, \dots, n$. The following definitions describe \mathcal{R} formally.

Definition 1. A scenario s is an assignment of costs to the binary variables, *i.e.*, a cost $c_i^s \in [l_i, u_i]$ is fixed for all x_i , $i = 1, \dots, n$.

Let \mathcal{S} be the set of all possible cost scenarios, which consists of the cartesian product of the continuous intervals $[l_i, u_i]$, $i = 1, \dots, n$. The cost of a solution $x \in \Omega$ in a scenario $s \in \mathcal{S}$ is given by $c^s x = \sum_{i=1}^n c_i^s x_i$.

Definition 2. A solution $opt(s) \in \Omega$ is said to be optimal for a scenario $s \in \mathcal{S}$ if it has the smallest cost in s among all the solutions in Ω , i.e., $opt(s) = \arg \min_{x \in \Omega} c^s x$.

Definition 3. The regret (robust deviation) of a solution $x \in \Omega$ in a scenario $s \in \mathcal{S}$, denoted by r_x^s , is the difference between the cost of x in s and the cost of $opt(s)$ in s , i.e., $r_x^s = c^s x - c^s opt(s)$.

Definition 4. The robustness cost of a solution $x \in \Omega$, denoted by R_x , is the maximum regret of x among all possible scenarios, i.e., $R_x = \max_{s \in \mathcal{S}} r_x^s$.

Definition 5. A solution $x^* \in \Omega$ is said to be robust if it has the smallest robustness cost among all the solutions in Ω , i.e., $x^* = \arg \min_{x \in \Omega} R_x$.

Definition 6. The interval min-max regret problem \mathcal{R} consists in finding a robust solution $x^* \in \Omega$.

For each scenario $s \in \mathcal{S}$, let $\mathcal{G}(s)$ denote the corresponding problem \mathcal{G} under cost vector $c^s \in \mathbb{R}_+^n$, i.e., the problem of finding an optimal solution $opt(s)$ for s . Also consider y , an n -dimensional vector of binary variables. Then, \mathcal{R} can be generically modeled as follows.

$$(\mathcal{R}) \quad \min \max_{s \in \mathcal{S}} \left(c^s x - \overbrace{\min_{y \in \Omega} c^s y}^{(\mathcal{G}(s))} \right) \quad (9)$$

$$s.t. \quad x \in \Omega. \quad (10)$$

The basic result presented below has been explicitly proved for several interval min-max regret problems (see, e.g., [12, 18, 23]) and generalized for the case of interval 0–1 min-max regret problems [1] (also see [2]).

Proposition 1 (Aissi et al. [1]). The regret of any feasible solution $x \in \Omega$ is maximum in the scenario $s(x)$ induced by x , defined as follows:

$$\text{for all } i \in \{1, \dots, n\}, \quad c_i^{s(x)} = \begin{cases} u_i, & \text{if } x_i = 1, \\ l_i, & \text{if } x_i = 0. \end{cases}$$

From Proposition 1, \mathcal{R} can be rewritten as

$$(\tilde{\mathcal{R}}) \quad \min \left(c^{s(x)} x - \min_{y \in \Omega} c^{s(x)} y \right) \quad (11)$$

$$s.t. \quad x \in \Omega. \quad (12)$$

One may note that the inner minimization in (11) does not define an LP problem, but a BILP one. Since, in general, there is no guarantee of integrality in solving the linear relaxation of this problem, we cannot represent it by means of extreme points and extreme rays, as in a classical Benders' reformulation [4].

Alternatively, we reformulate $\tilde{\mathcal{R}}$ by adding a free variable ρ and linear constraints that explicitly bound ρ with respect to all the feasible solutions that y can represent. The resulting MILP formulation (see, *e.g.*, [1]) is provided from (13) to (16).

$$(\mathcal{F}) \quad \min \left(\sum_{i=1}^n u_i x_i - \rho \right) \quad (13)$$

$$s.t. \quad \rho \leq \sum_{i=1}^n (l_i + (u_i - l_i)x_i)\bar{y}_i \quad \forall \bar{y} \in \Omega, \quad (14)$$

$$x \in \Omega, \quad (15)$$

$$\rho \text{ free.} \quad (16)$$

Constraints (14) ensure that ρ does not exceed the value related to the inner minimization in (11). Note that, in (14), \bar{y} is a constant vector, one for each solution in Ω . These constraints are tight whenever \bar{y} is optimal for the classical counterpart problem \mathcal{G} in the scenario $s(x)$. Constraints (15) define the feasible region referred to the x variables, and constraint (16) gives the domain of the variable ρ . Notice that the feasibility of \mathcal{F} solely relies on the feasibility of the corresponding classical optimization problem \mathcal{G} . Thus, for simplicity, we assume that \mathcal{F} is feasible in the remainder of this work.

The number of constraints (14) corresponds to the number of feasible solutions in Ω . As the size of this region may grow exponentially with the number of binary variables, this formulation is particularly suitable to be handled by decomposition methods, such as the logic-based Benders' decomposition framework detailed below.

2.2 Logic-Based Benders' Algorithm

The logic-based Benders' algorithm here described relies on the fact that, since several of constraints (14) might be inactive at optimality, they can be generated on demand whenever they are violated. In this sense, given a set $\Gamma \subseteq \Omega$, $\Gamma \neq \emptyset$, consider the *relaxed robustness cost* metric defined as follows.

Definition 7. A solution $opt(s, \Gamma) \in \Gamma$ is said to be Γ -relaxed optimal for a scenario $s \in \mathcal{S}$ if it has the smallest cost in s among all the solutions in Γ , *i.e.*, $opt(s, \Gamma) = \arg \min_{x \in \Gamma} c^s x$.

Definition 8. The Γ -relaxed robustness cost of a solution $x \in \Omega$, denoted by R_x^Γ , is the difference between the cost of x in the scenario $s(x)$ induced by x and the cost of a Γ -relaxed optimal solution $opt(s(x), \Gamma)$ in $s(x)$, *i.e.*, $R_x^\Gamma = c^{s(x)}x - c^{s(x)}opt(s(x), \Gamma)$.

Proposition 2. For any $\Gamma \subseteq \Omega$, $\Gamma \neq \emptyset$, and any solution $x \in \Omega$, the Γ -relaxed robustness cost R_x^Γ of x gives a lower bound on the robustness cost R_x of x .

Proof. Consider a set $\Gamma \subseteq \Omega$, $\Gamma \neq \emptyset$, and a solution $x \in \Omega$. According to Proposition 1, the robustness cost of x is given by $R_x = r_x^{s(x)} = c^{s(x)}x - c^{s(x)}\text{opt}(s(x))$, where $\text{opt}(s(x))$ is an optimal solution for the scenario $s(x)$ induced by x . By definition, the Γ -relaxed robustness cost of x is given by $R_x^\Gamma = c^{s(x)}x - c^{s(x)}\text{opt}(s(x), \Gamma)$, where $\text{opt}(s(x), \Gamma)$ is a Γ -relaxed optimal solution for $s(x)$. Notice that $c^{s(x)}\text{opt}(s(x)) \leq c^{s(x)}x'$ for all $x' \in \Omega$, including $\text{opt}(s(x), \Gamma)$. Therefore,

$$R_x^\Gamma = c^{s(x)}x - c^{s(x)}\text{opt}(s(x), \Gamma) \leq c^{s(x)}x - c^{s(x)}\text{opt}(s(x)) = R_x. \quad (17)$$

□

Proposition 3. *If $\Gamma = \Omega$, then, for any solution $x \in \Omega$, it holds that $R_x^\Gamma = R_x$.*

Proof. Consider the set $\Gamma = \Omega$ and a solution $x \in \Omega$. In this case, a Γ -relaxed optimal solution $\text{opt}(s(x), \Gamma)$ for $s(x)$ is also an optimal solution $\text{opt}(s(x))$ for this scenario. Therefore, considering Proposition 1,

$$R_x^\Gamma = c^{s(x)}x - c^{s(x)}\text{opt}(s(x), \Gamma) = c^{s(x)}x - c^{s(x)}\text{opt}(s(x)) = R_x. \quad (18)$$

□

Definition 9. *A solution $\tilde{x}^* \in \Omega$ is said to be Γ -relaxed robust if it has the smallest Γ -relaxed robustness cost among all the solutions in Ω , i.e., $\tilde{x}^* = \arg \min_{x \in \Omega} R_x^\Gamma$.*

Considering the relaxed metric discussed above, we detail a logic-based Benders' algorithm to solve formulation \mathcal{F} , given by (13)–(16). The procedure is described in Algorithm 1. Let $\Omega^\psi \subseteq \Omega$ be the set of solutions $\bar{y} \in \Omega$ (Benders' cuts) available at an iteration ψ . Also let \mathcal{F}^ψ be a relaxed version of \mathcal{F} in which constraints (14) are replaced by

$$\rho \leq \sum_{i=1}^n (l_i + (u_i - l_i)x_i)\bar{y}_i \quad \forall \bar{y} \in \Omega^\psi. \quad (19)$$

Thus, the relaxed problem \mathcal{F}^ψ , called *master problem*, is defined by (13), (15), (16) and (19). One may observe that \mathcal{F}^ψ is precisely the problem of finding a Γ -relaxed robust solution, with $\Gamma = \Omega^\psi$.

Let ub^ψ keep the best upper bound found (until an iteration ψ) on the solution of \mathcal{F} . Notice that, at the beginning of Algorithm 1, Ω^1 contains the initial Benders' cuts available, whereas ub^1 keeps the initial upper bound on the solution of \mathcal{F} . In this case, $\Omega^1 = \emptyset$ and $ub^1 := +\infty$. At each iteration ψ , the algorithm obtains a solution by solving a corresponding master problem \mathcal{F}^ψ and seeks a constraint (14) that is most violated by this solution. Initially, no constraint (19) is considered, since $\Omega^1 = \emptyset$. An initialization step is then necessary to add at least one solution to Ω^1 , thus avoiding unbounded solutions during the first resolution of the master problem. To this end, it is computed an optimal solution for the worst-case scenario s_u , in which $c^{s_u} = u$ (Step I, Algorithm 1).

Algorithm 1. Logic-based Benders' algorithm.

Input: Cost intervals $[l_i, u_i]$ referred to x_i , $i = 1, \dots, n$.**Output:** (\bar{x}^*, R^*) , where \bar{x}^* is a robust solution for \mathcal{F} , and R^* is its corresponding robustness cost. $\psi := 1$; $ub^1 := +\infty$; $\Omega^1 := \emptyset$;**Step I. (Initialization)**Find an optimal solution $\bar{y}^1 = \text{opt}(s_u)$ for the worst-case scenario s_u ; $\Omega^1 := \Omega^1 \cup \{\bar{y}^1\}$;**Step II. (Master problem)**Solve the relaxed problem \mathcal{F}^ψ , obtaining a solution $(\bar{x}^\psi, \bar{\rho}^\psi)$;**Step III. (Separation subproblem)**Find an optimal solution $\bar{y}^\psi = \text{opt}(s(\bar{x}^\psi))$ for the scenario $s(\bar{x}^\psi)$ induced by \bar{x}^ψ and use it to compute $R_{\bar{x}^\psi}$, the robustness cost of \bar{x}^ψ ;**Step IV. (Stopping condition)** $lb^\psi := u\bar{x}^\psi - \bar{\rho}^\psi$;**if** $lb^\psi \geq R_{\bar{x}^\psi}$ **then** $\bar{x}^* := \bar{x}^\psi$; $R^* := R_{\bar{x}^\psi}$;Return (\bar{x}^*, R^*) ;**end****else** $ub^\psi := \min\{ub^\psi, R_{\bar{x}^\psi}\}$; $ub^{\psi+1} := ub^\psi$; $\Omega^{\psi+1} := \Omega^\psi \cup \{\bar{y}^\psi\}$; $\psi := \psi + 1$;

Go to Step II;

end

After the initialization step, the iterative procedure takes place. At each iteration ψ , the corresponding relaxed problem \mathcal{F}^ψ is solved (Step II, Algorithm 1), obtaining a solution $(\bar{x}^\psi, \bar{\rho}^\psi)$. Then, the algorithm checks if $(\bar{x}^\psi, \bar{\rho}^\psi)$ violates any constraint (14) of the original problem \mathcal{F} , *i.e.*, if there is a constraint (19) that should have been considered in \mathcal{F}^ψ and was not. For this purpose, it is solved a *separation subproblem* that computes $R_{\bar{x}^\psi}$ (the actual robustness cost of \bar{x}^ψ) by finding an optimal solution $\bar{y}^\psi = \text{opt}(s(\bar{x}^\psi))$ for the scenario $s(\bar{x}^\psi)$ induced by \bar{x}^ψ (see Step III, Algorithm 1). Notice that the separation subproblems involve solving a classical optimization problem $\mathcal{G}(\bar{x}^\psi)$, *i.e.*, problem \mathcal{G} , given by (6)–(8), in the scenario $s(\bar{x}^\psi)$.

Let $lb^\psi = u\bar{x}^\psi - \bar{\rho}^\psi$ be the value of the objective function in (13) related to the solution $(\bar{x}^\psi, \bar{\rho}^\psi)$ of the current master problem \mathcal{F}^ψ . Notice that, considering $\Gamma = \Omega^\psi$, lb^ψ corresponds to the Γ -relaxed robustness cost of \bar{x}^ψ . Thus, according to Proposition 2, lb^ψ gives a lower (dual) bound on the solution of \mathcal{F} . Moreover, since \bar{x}^ψ is a feasible solution in Ω , its robustness cost $R_{\bar{x}^\psi}$ gives an upper (primal) bound on the solution of \mathcal{F} . Accordingly, if lb^ψ reaches $R_{\bar{x}^\psi}$, the algorithm stops. Otherwise, ub^ψ and $ub^{\psi+1}$ are both set to the best upper bound found by the algorithm until the iteration ψ . In addition, a new constraint

(19) is generated from \bar{y}^ψ and added to $\mathcal{F}^{\psi+1}$ by setting $\Omega^{\psi+1} := \Omega^\psi \cup \{\bar{y}^\psi\}$ (see Step IV of Algorithm 1). Notice that the algorithm stops when the value $\bar{\rho}^\psi$ corresponds to the cost of $\bar{y}^\psi = \text{opt}(s(\bar{x}^\psi))$ in the scenario $s(\bar{x}^\psi)$, *i.e.*, the optimal solution for \mathcal{F}^ψ is also feasible (and, therefore, optimal) for the original problem \mathcal{F} . The convergence of the algorithm is ensured by Proposition 3 and the following results.

Lemma 1. *Every separation subproblem that arises during the execution of Algorithm 1 is feasible.*

Proof. Assuming \mathcal{F} feasible, we must have $\Omega \neq \emptyset$. This implies the existence of at least one feasible solution for every scenario $s \in S$, and, thus, any classical problem \mathcal{G} that arises while executing Algorithm 1 is feasible. \square

Proposition 4. *At each iteration $\psi \geq 1$ of Algorithm 1, if the stopping condition is not satisfied, then the resolution of the corresponding separation subproblem leads to a new solution $\bar{y}^\psi \in \Omega \setminus \Omega^\psi$.*

Proof. Consider an iteration $\psi \geq 1$ of Algorithm 1 and assume, by contradiction, that (I) the stopping condition is not satisfied, and (II) the resolution of the separation subproblem of the current iteration ψ does not lead to a solution in $\Omega \setminus \Omega^\psi$. Let $(\bar{x}^\psi, \bar{\rho}^\psi)$ be the solution obtained from the resolution of the corresponding master problem \mathcal{F}^ψ . From Lemma 1, the subproblem referred to iteration ψ is feasible, and, thus, its resolution leads to an optimal solution $\bar{y}^\psi = \text{opt}(s(\bar{x}^\psi)) \in \Omega$ for the scenario $s(\bar{x}^\psi)$ induced by \bar{x}^ψ . From assumption (I), we must have $lb^\psi < R_{\bar{x}^\psi}$. Considering Proposition 1 and letting $\Gamma = \Omega^\psi$, we have that $lb^\psi = R_{\bar{x}^\psi}^\Gamma$, and, moreover,

$$c^{s(\bar{x}^\psi)} \bar{x}^\psi - c^{s(\bar{x}^\psi)} \text{opt}(s(\bar{x}^\psi), \Gamma) = R_{\bar{x}^\psi}^\Gamma \quad (20)$$

$$= lb^\psi \quad (21)$$

$$< R_{\bar{x}^\psi} \quad (22)$$

$$= c^{s(\bar{x}^\psi)} \bar{x}^\psi - c^{s(\bar{x}^\psi)} \text{opt}(s(\bar{x}^\psi)), \quad (23)$$

where $\text{opt}(s(\bar{x}^\psi), \Gamma)$ is a Γ -relaxed optimal solution for $s(\bar{x}^\psi)$, and $\text{opt}(s(\bar{x}^\psi))$ is an optimal solution for $s(\bar{x}^\psi)$. From (20)–(23), we obtain

$$c^{s(\bar{x}^\psi)} \text{opt}(s(\bar{x}^\psi), \Gamma) > c^{s(\bar{x}^\psi)} \text{opt}(s(\bar{x}^\psi)). \quad (24)$$

Notice that, since \bar{y}^ψ is also an optimal solution for $s(\bar{x}^\psi)$, it follows, from (24), that

$$c^{s(\bar{x}^\psi)} \text{opt}(s(\bar{x}^\psi), \Gamma) > c^{s(\bar{x}^\psi)} \text{opt}(s(\bar{x}^\psi)) = c^{s(\bar{x}^\psi)} \bar{y}^\psi. \quad (25)$$

Nevertheless, as $\text{opt}(s(\bar{x}^\psi), \Gamma)$ is a Γ -relaxed optimal solution for $s(\bar{x}^\psi)$, and, from Lemma 1 and assumption (II), \bar{y}^ψ belongs to $\Omega^\psi = \Gamma$, we also have that

$$c^{s(\bar{x}^\psi)} \text{opt}(s(\bar{x}^\psi), \Gamma) \leq c^{s(\bar{x}^\psi)} \bar{y}^\psi, \quad (26)$$

which, considering (25), defines a contradiction. \square

Theorem 1. *Algorithm 1 solves problem \mathcal{F} at optimality within a finite number of iterations.*

Proof. As Ω is defined in terms of binary variables, it consists of a finite discrete set of solutions. Thus, the convergence of Algorithm 1 is guaranteed by Propositions 3 and 4. \square

3 Concluding Remarks

In this work, we presented the first formal proof of the finite convergence of state-of-the-art logic-based Benders' decomposition algorithms for a class of robust optimization problems, namely interval 0–1 min-max regret problems. These algorithms were generically described by means of a logic-based Benders' decomposition framework, which was proved to converge to an optimal solution in a finite number of iterations.

References

1. Aissi, H., Bazgan, C., Vanderpooten, D.: Min-max and min-max regret versions of combinatorial optimization problems: a survey. *Eur. J. Oper. Res.* **197**(2), 427–438 (2009)
2. Averbakh, I.: On the complexity of a class of combinatorial optimization problems with uncertainty. *Math. Program.* **90**(2), 263–272 (2001)
3. Averbakh, I.: Computing and minimizing the relative regret in combinatorial optimization with interval data. *Discrete Optim.* **2**(4), 273–287 (2005)
4. Benders, J.F.: Partitioning procedures for solving mixed-variables programming problems. *Numer. Math.* **4**(1), 238–252 (1962)
5. Coco, A.A., Júnior, J.C.A., Noronha, T.F., Santos, A.C.: An integer linear programming formulation and heuristics for the minmax relative regret robust shortest path problem. *J. Glob. Optim.* **60**(2), 265–287 (2014)
6. Codato, G., Fischetti, M.: Combinatorial Benders' cuts for mixed-integer linear programming. *Oper. Res.* **54**(4), 756–766 (2006)
7. Côté, J.F., Dell'Amico, M., Iori, M.: Combinatorial Benders' cuts for the strip packing problem. *Oper. Res.* **62**(3), 643–661 (2014)
8. Fischetti, M., Salvagnin, D., Zanette, A.: A note on the selection of Benders' cuts. *Math. Program.* **124**(1–2), 175–182 (2010)
9. Garey, M.R., Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York (1979)
10. Geoffrion, A.M.: Generalized Benders decomposition. *J. Optim. Theo. Appl.* **10**(4), 237–260 (1972)
11. Hooker, J., Ottosson, G.: Logic-based Benders decomposition. *Math. Program.* **96**(1), 33–60 (2003)
12. Karaslan, O.E., Pinar, M.Ç., Yaman, H.: The robust shortest path problem with interval data. Technical report, Bilkent University, Ankara, Turkey (2001)
13. Kasperski, A.: *Discrete Optimization with Interval Data: Minmax Regret and Fuzzy Approach (Studies in Fuzziness and Soft Computing)*. Springer, Berlin (2008)

14. Kouvelis, P., Yu, G.: *Robust Discrete Optimization and Its Applications*. Kluwer Academic Publishers, Boston (1997)
15. Magnanti, T.L., Wong, R.T.: Accelerating Benders decomposition: algorithmic enhancement and model selection criteria. *Oper. Res.* **29**(3), 464–484 (1981)
16. McDaniel, D., Devine, M.: A modified Benders' partitioning algorithm for mixed integer programming. *Manag. Sci.* **24**(3), 312–319 (1977)
17. Montemanni, R.: A Benders decomposition approach for the robust spanning tree problem with interval data. *Eur. J. Oper. Res.* **174**(3), 1479–1490 (2006)
18. Montemanni, R., Barta, J., Gambardella, L.M.: The robust traveling salesman problem with interval data. *Transp. Sci.* **41**(3), 366–381 (2007)
19. Montemanni, R., Gambardella, L.M.: The robust shortest path problem with interval data via Benders decomposition. *4OR* **3**(4), 315–328 (2005)
20. Pereira, J., Averbakh, I.: Exact and heuristic algorithms for the interval data robust assignment problem. *Comput. Oper. Res.* **38**(8), 1153–1163 (2011)
21. Pereira, J., Averbakh, I.: The robust set covering problem with interval data. *Ann. Oper. Res.* **207**(1), 217–235 (2013)
22. Spall, J.C.: *Introduction to Stochastic Search and Optimization: Estimation Simulation and Control*. Wiley, New York (2003)
23. Yaman, H., Karaşan, O.E., Pinar, M.Ç.: The robust spanning tree problem with interval data. *Oper. Res. Lett.* **29**(1), 31–40 (2001)