

# RADIUS OF ROBUST FEASIBILITY FOR MIXED-INTEGER PROBLEMS

FRAUKE LIERS<sup>1,2</sup>, LARS SCHEWE<sup>3</sup>, JOHANNES THÜRAUF<sup>1,2</sup>

ABSTRACT. For a mixed-integer linear problem (MIP) with uncertain constraints, the radius of robust feasibility (RRF) determines a value for the maximal “size” of the uncertainty set such that robust feasibility of the MIP can be guaranteed. The approaches for the RRF in the literature are restricted to continuous optimization problems. We first analyze relations between the RRF of a MIP and its continuous linear (LP) relaxation. In particular, we derive conditions under which a MIP and its LP relaxation have the same RRF. Afterward, we extend the notion of the RRF such that it can be applied to a large variety of optimization problems and uncertainty sets. In contrast to the setting commonly used in the literature, we consider for every constraint a potentially different uncertainty set that is not necessarily full-dimensional. Thus, we generalize the RRF to MIPs as well as to include “safe” variables and constraints, i.e., where uncertainties do not affect certain variables or constraints. In the extended setting, we again analyze relations between the RRF for a MIP and its LP relaxation. Afterward, we present methods for computing the RRF of LPs as well as of MIPs with safe variables and constraints. Finally, we show that the new methodologies can be successfully applied to the instances in the MIPLIB 2017 for computing the RRF.

## 1. INTRODUCTION

Robust optimization is a well-established method for protecting an optimization problem from data uncertainties that are usually defined via so-called uncertainty sets. Such data uncertainties may arise as a result of estimation and prediction errors as well as from a lack of (future) information. Robust optimization plays an important role in many applications such as finance, energy, supply chain, health care, etc., see [25] and the literature therein. For detailed overviews of the research area of robust optimization, we refer to [3, 5, 7, 10, 25]. One of the main goals consists in finding robust feasible solutions, i.e., solutions which are feasible for all realizations of a given uncertainty set. A solution is robust optimal if it is robust feasible and attains the best possible objective value. The corresponding robust optimization problem, also called robust counterpart, is, in general, semi-infinite. Nevertheless, for several important classes of optimization problems and uncertainty sets, it is possible to reformulate the robust counterpart as an algorithmically tractable finite optimization problem. This is in particular true for mixed-integer linear optimization and convex uncertainty sets, see for example [4] for a comprehensive treatment.

Intensive research has been conducted in developing algorithmically tractable robust counterparts. However, in applications it is also important to construct appropriate uncertainty sets. Some proposals for constructing “good” uncertainty sets are given in [6, 8, 25]. High-volume uncertainty sets may lead to overly conservative solutions that are overly protected and furthermore lead to bad objective function

---

*Date:* June 18, 2020.

*2010 Mathematics Subject Classification.* 90-XX, 90C31, 90C11.

*Key words and phrases.* Robust Optimization, Mixed-integer programming, Uncertainty sets, Robust feasibility.

values, when compared to the nominal solution. The overall goal of constructing “good” uncertainty sets consists in prohibiting too conservative, intractable, or even infeasible robust optimization problems due to the choice of the uncertainty set. In order to achieve these goals, it is useful to know the maximal “size” of a given uncertainty set such that a robust feasible solution still exists. In this paper, we study one notion of “size”: the radius of robust feasibility (RRF). It is motivated by the notion of the consistency radius used in the linear semi-infinite programming, see [11–13].

In this work, we investigate the problem of determining the RRF for a mixed-integer linear optimization problem (MIP), both from a theoretical as well as from a practical point of view. To evaluate our methods on a set of realistic MIPs from different applications, we apply them to compute the RRF for the benchmark instances of the MIPLIB 2017 library.

In general, the RRF is defined as the supremum over all scaled sizes of a given uncertainty set such that robust feasibility is guaranteed. Consequently, it is possible that the supremum is not attained, i.e., the RRF is not attained. In this case, the uncertain problem is not feasible for the uncertainty set scaled by the RRF, but it is feasible for every smaller scaling, see [23]. The RRF has been researched only for continuous problems. For linear problems (LPs), theoretical and numerical tractable models for the RRF w.r.t. different compact and convex uncertainty sets are provided in [19, 23, 24]. The RRF is introduced in robust convex optimization in [22]. The authors further provide an upper bound of the RRF for convex problems with convex polynomial constraints and establish a method for computing the RRF of convex problems with SOS-convex polynomial constraints. In [18, 29], exact analytical formulas for the RRF of convex problems with general convex and compact uncertainty sets are established. We also note that in the recent paper [18] lower and upper bounds for the RRF of convex problems with different full-dimensional uncertainty sets for every constraint are given. We note that the RRF has connections to recent developments in the fields of stability and sensitivity analysis of robust optimization problems, see e.g., [16, 20], because it computes the solution that is most insensitive w.r.t. feasibility and changes, in form of scaling, of the uncertainty set.

We generalize the above-mentioned approaches in three directions: We allow that the uncertainty sets are different and not necessarily full-dimensional for every constraints. We do not require zero to be in the interior of the uncertainty set and finally, we allow integer variables in the optimization problem. This enables us to consider a wider variety of applications for the RRF. For instance, we can include “safe” constraints and variables, i.e., constraints and variables that are not affected by uncertainty. For example, if all coefficients of a constraint are deterministic, this constraint is safe. If all coefficients of some variable are deterministic in the constraint system, this variable is safe. The drawback of this generalization is, that we lose some of the nice theoretical properties of the RRF, e.g., finiteness, see [23]. Furthermore, the generalizations require the development of new algorithmic techniques to compute the RRF.

The RRF has been studied for specific applications. For instance, in [14] the authors try to find the “most robust” facilities w.r.t. demand uncertainties for the Weber problem of facility location design. One can show that their problem is equivalent to computing the RRF. However, for this equivalence to hold, one cannot use the standard definition of the RRF, but one needs to extend it to include safe constraints and variables as in Section 3. The core idea in [14] is to remove the objective of the original problem and reintroduce it into the problem as a budget constraint for a fixed budget (e.g. the original optimal value). Then one can compute

the RRF, with the budget constraint considered safe, to obtain a solution that can be seen as a “most robust” solution. The budget specifies how much a decision maker is willing to pay to obtain a robust solution. With the help of varying the budget, one can support decision makers by showing them the trade-off between robustness and worse objective values. With our work, the same idea can now be applied to general MIPs. The authors of [15] also use the concept of the RRF in facility location design. They consider a bi-objective problem that consists of maximizing robustness via the RRF and minimizing the estimated total cost. For LPs another example is the flexibility index problem, e.g., the RRF of an LP with a box uncertainty set, see [33].

A more complex variant of the RRF plays an important role in the context of design and control of gas networks. In the European Entry-Exit market system, the transmission system operator is obliged to allocate so called technical capacities in the network while guaranteeing the feasibility of the gas transport for any injection and withdrawal within these capacities, see [28] for a more detailed explanation. The computation of technical capacities leads to a two-stage nonlinear robust optimization problem that has not been solved in general so far. The latter problem can be solved by applying a complex variant of the extended RRF including “safe” constraints and variables and different radii for different constraints. Thus, this work is a first step towards computing technical capacities in gas network operations.

The key contributions of our paper are as follows:

- (i) We first introduce the RRF for MIPs in Section 2. We then analyze in detail the relations between the RRF of a MIP and its LP relaxation in the common setting of the literature, i.e., where the uncertainty set is full-dimensional. We prove the main result that if the RRF of the LP relaxation is not attained, then this RRF equals the RRF of the corresponding MIP. The latter result enables us to compute the RRF of a MIP using known techniques for the RRF of LPs under certain conditions.
- (ii) We extend the concept of the RRF to include “safe” variables and constraints in Section 3 in order to make the RRF applicable to a broader spectrum of problems and applications. We then again analyze relations between the RRF of a MIP and its LP relaxation. Further, we prove a necessary optimality condition for the RRF which is also sufficient under additional assumptions.
- (iii) We provide first algorithms for computing the RRF including “safe” variables and constraints in Section 4. Finally, we present a computational study of the RRF w.r.t. the MIPLIB 2017 library, see [30]. We compare the performance of the proposed methods and the computed RRF. We also consider the *price of robustness* which measures the difference between the optimal objective value of the nominal problem and the corresponding value of the robust problem and discuss the obtained results.

## 2. RELATIONS BETWEEN THE RRF OF A MIP AND OF ITS LP RELAXATION

In this section, we first introduce the radius of robust feasibility for a MIP and then relate it to that of its LP relaxation. In the following, let us consider a feasible MIP with coefficients  $\bar{a}^j \in \mathbb{R}^n$  and  $\bar{b}^j \in \mathbb{R}$ ,  $j \in J$ , of the constraints and finite index set  $J \subset \mathbb{N}$  that is composed of

$$\min_{x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}} \{c^T x : (\bar{a}^j)^T x \leq \bar{b}^j, j \in J\}. \quad (\text{P})$$

For fixed  $\alpha \geq 0$ , let the robust counterpart for the uncertain MIP  $(\mathbf{P})$  with uncertainty set  $\alpha\mathcal{U} \subseteq \mathbb{R}^{n+1}$  be given by

$$\min_{x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}} \{c^T x : (a^j)^T x \leq b^j, \forall (a^j, b^j) \in \{(\bar{a}^j, \bar{b}^j) + \alpha u : u \in \mathcal{U}\}, j \in J\}, \quad (\text{PR}_\alpha)$$

whereby  $\mathcal{U} \subseteq \mathbb{R}^{n+1}$  is a convex and compact uncertainty set. In analogy to [19, 22, 23, 29], we further assume that the uncertainty set  $\mathcal{U}$  contains zero in its interior.

*Assumption 1.* Uncertainty set  $\mathcal{U}$  includes zero in its interior, i.e.,  $0 \in \text{int } \mathcal{U}$  holds.

This implies that the uncertainty set  $\mathcal{U}$  is full-dimensional, i.e. every variable is affected by uncertainties. We note that one can transform a robust optimization problem with an uncertainty set that does not contain zero to an equivalent robust problem with an uncertainty set that *contains* zero, see Chapter 1 of the book [3]. It is, however, not possible to guarantee that zero is *in the interior* of the uncertainty set  $\mathcal{U}$ . This is the case, for instance, if one variable is not affected by uncertainty, i.e., the projection of  $\mathcal{U}$  on a single variable is just the set containing only zero. Furthermore, we note that the standard transformation of the uncertainty set does not maintain the form of  $(\text{PR}_\alpha)$ .

Assumption 1 guarantees that the radius of robust feasibility of LPs is finite, as shown in [23]. Additionally, we assume that the nominal problem  $(\text{PR}_0)$  is feasible, i.e.,  $\{x \in \mathbb{Z}^k \times \mathbb{R}^{n-k} : \bar{a}^j x \leq \bar{b}^j, j \in J\} \neq \emptyset$ . Following the notion of [19, 23], who consider the radius of robust feasibility for linear problems, we define the *radius of robust feasibility* (RRF) for the parametric uncertain mixed-integer problem  $(\mathbf{P})$  as

$$\rho_{\text{MIP}} := \sup\{\alpha \geq 0 : (\text{PR}_\alpha) \text{ is feasible}\}.$$

The definition of the RRF  $\rho_{\text{MIP}}$  does not necessarily imply the feasibility of  $(\text{PR}_{\rho_{\text{MIP}}})$ , even in the case of linear problems, see Example 2.2 in [23]. If  $(\text{PR}_{\rho_{\text{MIP}}})$  is feasible, we say that the RRF is attained, otherwise it is not attained. Proposition 2.3 in [23] states a sufficient condition so that the RRF is attained by a feasible solution.

We note that  $(\text{PR}_\alpha)$  is a semi-infinite MIP that consists of infinitely many constraints and finitely many variables. Thus, it cannot easily be solved by known techniques. We now reformulate  $(\text{PR}_\alpha)$  with the help of Fenchel duality in order to obtain an ordinary robust counterpart, i.e., the robust counterpart consists of finitely many variables and constraints. For ease of notation, we use index set  $I := \{1, \dots, n\}$  and  $b := n + 1$  in the remainder of this paper. We further introduce the indicator function  $\delta(x \mid \mathcal{U})$  for  $x \in \mathbb{R}^{n+1}$ , which evaluates to zero if  $x \in \mathcal{U}$  holds and otherwise to  $+\infty$ . Moreover, let  $\delta^*(y \mid \mathcal{U}) = \sup_{u \in \mathcal{U}} y^T u$  denote the conjugate function of the indicator function, which is also called support function.

**Proposition 2.1.** *Let  $\alpha \geq 0$  be fixed. Then, the feasible region of  $(\text{PR}_\alpha)$  equals the feasible region of the ordinary counterpart*

$$\{x \in \mathbb{Z}^n \times \mathbb{R}^{n-k} \mid (\bar{a}^j)^T x + \alpha \delta^*((x, -1)^T \mid \mathcal{U}) \leq \bar{b}^j, j \in J\}. \quad (1)$$

*Proof.* The claim follows from Theorem 2 in [4] and the positive homogeneity of  $\delta^*(y \mid \mathcal{U})$ .  $\square$

Consequently, we obtain

$$\min_{x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}} \{c^T x : (\bar{a}^j)^T x + \alpha \delta^*((x, -1)^T \mid \mathcal{U}) \leq \bar{b}^j, j \in J\}. \quad (\text{PRC}_\alpha)$$

as the ordinary robust counterpart of  $(\text{PR}_\alpha)$ .

In general, for fixed  $\alpha \geq 0$ , problem  $(\text{PRC}_\alpha)$  is a convex constrained mixed-integer problem. This holds, because for a convex and compact set  $\mathcal{U}$  the support function  $\delta^*(y \mid \mathcal{U})$  is convex in  $y$ , see [9]. For many uncertainty sets  $\mathcal{U}$  such as boxes,

balls, cones, polyhedrals, or convex functions, an explicit formulation of  $(\text{PRC}_\alpha)$ , especially the computation of the support function, can be found in [4].

All existing techniques for computing the RRF of continuous problems, such as [18, 19, 22, 23, 29], are based on concepts that are not transferable to MIPs. Hence, we now analyze relations about the RRF of  $(\text{P})$  and of its LP relaxation

$$\min_{x \in \mathbb{R}^n} \{c^T x : (\bar{a}^j)^T x \leq \bar{b}^j, j \in J\}. \quad (\text{LP})$$

The robust counterpart for the uncertain linear problem  $(\text{LP})$  with uncertainty set  $\alpha\mathcal{U} \subseteq \mathbb{R}^{n+1}$  and its ordinary reformulation equal the continuous relaxations of  $(\text{PR}_\alpha)$  and  $(\text{PRC}_\alpha)$ . We denote them by

$$\min_{x \in \mathbb{R}^n} \{c^T x : (a^j)^T x \leq b^j, \forall (a^j, b^j) \in \{(\bar{a}^j, \bar{b}^j) + \alpha u : u \in \mathcal{U}\}, j \in J\}, \quad (\text{LPR}_\alpha)$$

$$\min_{x \in \mathbb{R}^n} \{c^T x : (\bar{a}^j)^T x + \alpha \delta^*((x, -1)^T | \mathcal{U}) \leq \bar{b}^j, j \in J\}. \quad (\text{LPRC}_\alpha)$$

We first prove some basic results, which show among other things that the RRF of  $(\text{LP})$  is always an upper bound for  $\rho_{\text{MIP}}$ .

**Theorem 2.2.** *Let  $\rho_{\text{MIP}}$  be the RRF of  $(\text{P})$ . The RRF of its continuous relaxation  $(\text{LP})$  is denoted by  $\rho_{\text{LP}}$ . Then, the following statements hold:*

- (i)  $0 \leq \rho_{\text{MIP}} \leq \rho_{\text{LP}}$ .
- (ii) RRF  $\rho_{\text{MIP}}$  is finite.

*Proof.* Each feasible solution of a MIP is feasible for its LP relaxation. Thus, the RRF of  $(\text{LP})$  is always an upper bound for the corresponding RRF of  $(\text{P})$ . Furthermore, the RRF of an LP is finite if  $0 \in \text{int } \mathcal{U}$  holds, see [23]. Consequently, the RRF of  $(\text{P})$  is finite as well.  $\square$

Next, we state a monotonicity fact regarding the feasibility of  $(\text{PR}_\alpha)$ . It is based on the observation that if a robust optimization problem is feasible, then so is the same problem for a subset of the uncertainty set.

*Observation 2.3.* If  $x$  is a feasible solution to  $(\text{PR}_\alpha)$ , then  $x$  is also feasible for  $(\text{PR}_{\alpha'})$  for all  $\alpha' \in [0, \alpha]$ .

We now show with the help of an example that the RRF of  $(\text{P})$  and of its LP relaxation  $(\text{LP})$  are not necessarily equal.

**Example 2.4.** The constraints of the nominal problem are given by

$$-2x_1 \leq -1.5, \quad 2x_1 \leq 3.5, \quad x_1 \in \mathbb{Z}, \quad (2)$$

with uncertainty set  $\mathcal{U} := [-1, 1]^2$ . Since

$$\delta^*((x_1, -1) | [-1, 1]^2) = \max_{u_1, u_2 \in [-1, 1]} (u_1 x_1 - u_2) = |x_1| + 1$$

holds, Proposition 2.1 leads to the following robust counterpart of (2)

$$-2x_1 + \alpha|x_1| \leq -1.5 - \alpha, \quad 2x_1 + \alpha|x_1| \leq 3.5 - \alpha, \quad x_1 \in \mathbb{Z}. \quad (3)$$

The only feasible solution for the nominal problem (2) is  $x_1 = 1$ . Further,  $x_1 = 1$  is feasible to (3) if and only if  $\alpha \in [0, 0.25]$  holds. Consequently, the RRF of (2) equals 0.25. Now, we consider the LP relaxation of (2). The corresponding robust counterpart equals the relaxation of (3). If  $x_1$  is a feasible solution of the relaxation of (3), then it is a feasible solution of (2). Consequently,  $0.75 \leq x_1 \leq 1.75$  holds. Since  $x_1 > 0$ , one has

$$\frac{1.5 + \alpha}{2 - \alpha} \leq x_1 \leq \frac{3.5 - \alpha}{\alpha + 2}$$

which entails  $\alpha \leq \frac{4}{9}$ . Conversely, if  $0 \leq \alpha \leq \frac{4}{9}$ , then every  $x_1$  satisfying the previous inequalities is a feasible solution of the relaxation of (3). Thus, the RRF of the

relaxation is  $\frac{4}{9}$  that is attained by  $x_1 = 1.25$ . We note that the RRF of (2) and its relaxation are attained, i.e.,  $(\text{PR}_{0.25})$  and  $(\text{LPR}_{\frac{4}{9}})$  are feasible.

This leads to the following.

*Observation 2.5.* Let  $\rho_{\text{MIP}}$  be the RRF of (P) and  $\rho_{\text{LP}}$  the RRF of its LP relaxation (LP). MIPs with  $\rho_{\text{MIP}} < \rho_{\text{LP}}$  exist.

We now state the main result of this section, namely that if  $\rho_{\text{LP}}$  is not attained, then (P) and (LP) have the same RRF. This result provides sufficient conditions such that the RRF of a MIP can be computed by the RRF of the LP relaxation. In detail, we first compute the RRF of the LP relaxation with known techniques. If this RRF is not attained, then it is also the RRF of the corresponding MIP. Otherwise, we obtain an upper bound which is useful for computing the RRF as we will see in Section 4. Additionally, we show that a similar connection between the RRF of a MIP and its LP relaxation is not necessarily given if  $\rho_{\text{LP}}$  is attained. These findings are summarized in the next theorem.

**Theorem 2.6.** *Let  $\rho_{\text{MIP}}$  be the RRF of (P) and  $\rho_{\text{LP}}$  the RRF of its LP relaxation (LP). Then, the following statements hold:*

- (i) *If the RRF of (LP) is not attained, then  $\rho_{\text{MIP}} = \rho_{\text{LP}}$  holds.*
- (ii) *If the RRF of (P) is attained, then the RRF of (LP) is also attained.*
- (iii) *MIPs exist such that the RRF  $\rho_{\text{LP}}$  is attained and  $\rho_{\text{MIP}}$  is attained.  
MIPs exist such that the RRF  $\rho_{\text{LP}}$  is attained and  $\rho_{\text{MIP}}$  is not attained.*

We will now present several examples and lemmas. With their help, we will prove Theorem 2.6 at the end of this section.

In general, the RRF is not necessarily attained by a feasible solution. If (P) and (LP) have the same RRF and  $(\text{PR}_{\rho_{\text{MIP}}})$  is feasible, then the RRF of (LP) is also attained because each feasible solution of  $(\text{PR}_{\rho_{\text{MIP}}})$  is also feasible to  $(\text{LPR}_{\rho_{\text{LP}}})$ . A reversal of this relation is not true in general. That means, if the RRF of (LP) is attained, then  $(\text{PR}_{\rho_{\text{MIP}}})$  is not necessarily feasible. We show this with the help of the following example.

**Example 2.7.** The constraints of the nominal problem are given by

$$-x_1 - 2x_2 \leq 0.5, \quad -x_1 + 2x_2 \leq 2.5, \quad x_1, x_2 \in \mathbb{Z}, \quad (4)$$

with uncertainty set  $\mathcal{U} := [-1, 1]^3$ . From Proposition 2.1 the robust counterpart of (4) reads as

$$-x_1 + \alpha|x_1| - 2x_2 + \alpha|x_2| \leq 0.5 - \alpha, \quad (5a)$$

$$-x_1 + \alpha|x_1| + 2x_2 + \alpha|x_2| \leq 2.5 - \alpha, \quad x_1, x_2 \in \mathbb{Z}. \quad (5b)$$

For  $\alpha \in [0, 1)$ , we set  $x_2 = 0$  and  $(-1 + \alpha) < 0$  holds. Consequently,  $(x_1, 0)$  is feasible for (5) whenever  $x_1 \in \mathbb{N}$  satisfies  $x_1 \geq \frac{0.5 - \alpha}{\alpha - 1}$ . Thus, the RRF of (4) is at least 1. We now consider  $\alpha = 1$ . Since  $-x_1 + \alpha|x_1| \geq 0$  holds for every  $x_1 \in \mathbb{R}$ , it follows  $x_2 > 0$  by (5a). Consequently, from (5) we obtain  $-x_2 \leq -0.5$  and  $3x_2 \leq 1.5$  that has to be satisfied by an integer solution which leads to a contradiction. Consequently, the RRF of (4) equals 1 due to Observation 2.3 and further it is not attained. We now consider the relaxation of (4), i.e.,  $x_1, x_2 \in \mathbb{R}$ . Its robust counterpart equals the relaxation of (5). Then,  $x_1 = 0, x_2 = 0.5$  is a feasible solution for  $\alpha \in [0, 1]$  of the corresponding robust counterpart. For  $\alpha > 1$  the inequality  $-x_1 + \alpha|x_1| \geq 0$  holds for  $x_1 \in \mathbb{R}$  and thus, from (5a) it follows  $x_2 > 0$ . Consequently, we obtain from (5) the inequalities  $-x_2 < -0.5$  and  $x_2 < 0.5$  that have to be satisfied, which is a contradiction. Thus, the RRF of the relaxation of (4) equals 1 and it is attained.

If we slightly increase the right-hand side of (4), then we obtain the same result that  $(\text{LPR}_{\rho_{\text{LP}}})$  is feasible and  $(\text{PR}_{\rho_{\text{MIP}}})$  is infeasible, but this time  $\rho_{\text{MIP}} < \rho_{\text{LP}}$  holds. An example for this adaption is given as follows.

**Example 2.8.** The constraints of the nominal problem are given by

$$-x_1 - 2x_2 \leq 0.6, \quad -x_1 + 2x_2 \leq 2.6, \quad x_1, x_2 \in \mathbb{Z}.$$

Then, the RRF of (P) equals 1 and it is not attained. The RRF of (LP) equals  $\frac{16}{15}$  that is attained by  $x_1 = 0, x_2 = 0.5$ .

We now show several statements that lead to the proof of the main result (i) of Theorem 2.6. The latter says that if (LP) does not attain its RRF, then (P) and (LP) have the same RRF. We first prove that if the RRF  $\rho_{\text{MIP}}$  is not attained, then an unbounded sequence of solutions exists such that for every  $\alpha < \rho_{\text{MIP}}$  an element of the sequence solves  $(\text{PRC}_{\alpha})$ .

**Lemma 2.9.** *If the RRF  $\rho_{\text{MIP}}$  of (P) is not attained, then a positive and strictly increasing sequence  $(\alpha^l)_{l \in \mathbb{N}}$  and an unbounded sequence in  $\mathbb{R}^n$ ,  $(x^l)_{l \in \mathbb{N}}$ , exist such that  $(\alpha^l)_{l \in \mathbb{N}}$  converges to  $\rho_{\text{MIP}}$  and  $x^l$  is feasible to  $(\text{PRC}_{\alpha^l})$  for all  $l \in \mathbb{N}$ .*

*Proof.* Since (P) is feasible and  $\rho_{\text{MIP}}$  is not attained,  $\rho_{\text{MIP}} > 0$  holds. Consequently, a positive and strictly increasing sequence  $(\alpha^l)_{l \in \mathbb{N}}$  that converges to  $\rho_{\text{MIP}}$  exists. Furthermore, a sequence in  $\mathbb{R}^n$ ,  $(x^l)_{l \in \mathbb{N}}$ , exists such that  $x^l$  is feasible to  $(\text{PRC}_{\alpha^l})$  for all  $l \in \mathbb{N}$ . We now have to show that the sequence  $(x^l)_{l \in \mathbb{N}}$  is unbounded. To this end, we contrarily assume that  $(x^l)_{l \in \mathbb{N}}$  is bounded. Consequently, and by passing to a subsequence if necessary, we may assume that  $x^l \rightarrow \bar{x}$  holds, with  $\bar{x}_i \in \mathbb{Z}$  for  $i = 1, \dots, k$  thanks to the closedness of  $\mathbb{Z}$ . Considering  $(\text{PRC}_{\alpha^l})$  together with a solution  $x^l$  for an arbitrary  $j \in J$  leads to

$$(\bar{a}^j)^T x^l + \alpha^l \delta^*((x^l, -1)^T | \mathcal{U}) \leq \bar{b}^j. \quad (6)$$

Passing to the limit in (6), we obtain

$$(\bar{a}^j)^T \bar{x} + \rho_{\text{MIP}} \delta^*((\bar{x}, -1)^T | \mathcal{U}) \leq \bar{b}^j.$$

Thus,  $\bar{x}$  is a feasible solution to  $(\text{PRC}_{\rho_{\text{MIP}}})$ , which contradicts the requirements.  $\square$

We next prove that under the given conditions we can arbitrarily expand the slack of any constraint of  $(\text{PR}_{\alpha^l})$ .

**Lemma 2.10.** *Let  $(\alpha^l)_{l \in \mathbb{N}}$  be a strictly increasing positive sequence and an unbounded sequence in  $\mathbb{R}^n$ ,  $(x^l)_{l \in \mathbb{N}}$ , exist such that  $x^l$  is feasible to  $(\text{PR}_{\alpha^l})$  for all  $l \in \mathbb{N}$ . Then, for an arbitrary value  $M \geq 0$  and index  $\hat{l} \in \mathbb{N}$  there exists an index  $\bar{l} > \hat{l}$  such that for all  $u = \begin{pmatrix} u_a \\ u_b \end{pmatrix} \in \mathcal{U}$ ,  $j \in J$ , and  $l \geq \bar{l}$  the inequality*

$$(\bar{a}^j)^T x^l + \alpha^l (u_a^T x^l - u_b) + M \leq \bar{b}^j$$

*holds.*

*Proof.* For a sufficiently small number  $\beta > 0$ , we know that  $\pm \beta e_v \in \mathcal{U}$  for  $v \in I$  holds, whereby  $e_v$  is the  $v$ th unit vector of  $\mathbb{R}^{n+1}$ , because  $0 \in \text{int } \mathcal{U}$ . Passing to a subsequence if necessary, we know that  $v \in I$  with  $|x_v^l| \rightarrow +\infty$  exists because the sequence  $(x^l)_{l \in \mathbb{N}}$  is unbounded. We can assume w.l.o.g. that  $x_v^l \rightarrow +\infty$  holds due to  $\pm \beta e_v \in \mathcal{U}$ . Additionally, we can assume that  $(x_v^l)_{l > \hat{l}}$  is strictly increasing and we know that  $(\alpha^l)_{l \in \mathbb{N}}$  is strictly increasing. Consequently, an index  $\bar{l}$  exists such that for all  $l \geq \bar{l}$  the inequality  $(\alpha^l - \alpha^{\hat{l}}) \beta x_v^l \geq M$  holds. We now choose an arbitrary index  $j \in J$ ,  $u \in \mathcal{U}$ , and consider  $l > \bar{l}$ . From the convexity of  $\mathcal{U}$  it follows  $u' := \frac{\alpha^{\hat{l}}}{\alpha^l} u + (1 - \frac{\alpha^{\hat{l}}}{\alpha^l}) \beta e_v \in \mathcal{U}$ . The element  $x^l$  is a feasible solution to  $(\text{PR}_{\alpha^l})$ . Hence,

$$(\bar{a}^j)^T x^l + \alpha^l ((u')^T x^l - u'_b) = (\bar{a}^j)^T x^l + \alpha^{\hat{l}} (u_a^T x^l - u_b) + (\alpha^l - \alpha^{\hat{l}}) \beta x_v^l \leq \bar{b}^j \quad (7)$$

is satisfied for every  $l > \bar{l}$ . The inequality  $(\alpha^l - \alpha^{\hat{l}})\beta x_v^l \geq M$ , which is independent from the chosen  $u$ , holds and thus, (7) shows the claim.  $\square$

Due to the compactness of  $\mathcal{U}$ , we know that rounding down any solution leads to a bounded difference in the left side of any constraint in  $(\text{PRC}_\alpha)$ . For  $x \in \mathbb{R}^n$ ,  $\lfloor x \rfloor$  denotes the vector whose  $v$ th component is the lower integer part of  $x_v$ .

**Lemma 2.11.** *For fixed  $\alpha \geq 0$ , a positive value  $M > 0$  exists such that the inequalities*

$$|(\bar{a}^j)^T(x - \lfloor x \rfloor) + \alpha u_j^T(x - \lfloor x \rfloor)| \leq M \quad (8)$$

are satisfied for any  $u \in \mathcal{U}$ ,  $x \in \mathbb{R}^n$ , and  $j \in J$ .

Finally, we can prove Theorem 2.6.

*Proof of Theorem 2.6.* Examples 2.4, 2.7, and 2.8 show Statement (iii).

We now prove Statement (i). To this end, we assume w.l.o.g. that  $\rho_{\text{LP}}$  is positive. The RRF  $\rho_{\text{LP}}$  is not attained and hence, from Lemma 2.9 it follows that a strictly increasing positive sequence  $(\alpha^l)_{l \in \mathbb{N}}$  with  $0 < \alpha^l < \rho_{\text{LP}}$  and an unbounded sequence in  $\mathbb{R}^n$ ,  $(x^l)_{l \in \mathbb{N}}$ , exist such that  $\alpha^l$  converges to  $\rho_{\text{LP}}$  and  $x^l$  is feasible to  $(\text{LPR}_{\alpha^l})$  for all  $l \in \mathbb{N}$ . For an arbitrary fixed index  $\hat{l} \in \mathbb{N}$ , we now construct a solution  $\hat{x}^{\hat{l}}$  that is feasible to  $(\text{PR}_{\alpha^{\hat{l}}})$ . Due to Lemma 2.11, we can choose  $M > 0$  such that (8) is satisfied. We now apply Lemma 2.10 for this value  $M$ . Consequently, a solution  $x^l$  exists such that the inequalities

$$(\bar{a}^j)^T x^l + \alpha^{\hat{l}}(u_j^T x^l - u_b) + M \leq \bar{b}^j, \quad u \in \mathcal{U}, j \in J, \quad (9)$$

are satisfied. For an arbitrary element  $u \in \mathcal{U}$  and  $j \in J$ , the inequalities

$$\begin{aligned} \bar{b}^j &\geq (\bar{a}^j)^T x^l + \alpha^{\hat{l}}(u_j^T x^l - u_b) + M \\ &= (\bar{a}^j)^T \lfloor x^l \rfloor + \alpha^{\hat{l}}(u_j^T \lfloor x^l \rfloor - u_b) + (\bar{a}^j)^T (x^l - \lfloor x^l \rfloor) + \alpha^{\hat{l}} u_j^T (x^l - \lfloor x^l \rfloor) + M \\ &\geq (\bar{a}^j)^T \lfloor x^l \rfloor + \alpha^{\hat{l}}(u_j^T \lfloor x^l \rfloor - u_b), \end{aligned}$$

follow from (8) and (9). Thus,  $\hat{x}^{\hat{l}} := \lfloor x^l \rfloor$  is an integer solution to  $(\text{PR}_{\alpha^{\hat{l}}})$ . We have arbitrarily chosen  $\hat{l} \in \mathbb{N}$  and hence, we can construct for each  $\hat{l} \in \mathbb{N}$  an integer solution which is feasible for  $(\text{PR}_{\alpha^{\hat{l}}})$ . This, the convergence of  $(\alpha^{\hat{l}})_{\hat{l} \in \mathbb{N}}$  to  $\rho_{\text{LP}}$ , and Statement (i) of Theorem 2.2, prove that  $\rho_{\text{MIP}} = \rho_{\text{LP}}$  holds.

We now show Statement (ii). We contrarily assume that the RRF of  $(\text{LP})$  is not attained. Thus,  $\rho_{\text{MIP}} = \rho_{\text{LP}}$  follows from Statement (i) of Theorem 2.6. Due to the requirements,  $(\text{PR}_{\rho_{\text{MIP}}})$  is feasible, which is a contradiction to the assumption, because each feasible solution of  $(\text{PR}_{\rho_{\text{MIP}}})$  is feasible to  $(\text{LPR}_{\rho_{\text{LP}}})$ .  $\square$

The proof of this theorem closes the section. We will extend our investigations to linear optimization problems that contain safe constraints and variables in the following section.

### 3. EXTENSION OF THE RRF TO INCLUDE SAFE CONSTRAINTS AND VARIABLES

As mentioned in the introduction, there is a need to integrate safe variables and constraints into the concept of the RRF since often only parts of optimization models are affected by uncertainty in practice. Thus, a full-dimensional uncertainty set  $\mathcal{U}$  with  $0 \in \text{int } \mathcal{U}$  such as in Assumption 1 is not given in this context. Consequently, many known techniques for computing the RRF of LPs such as in [18, 19, 22, 23, 29] are not applicable anymore. Moreover, it is sometimes necessary to choose different not necessarily full-dimensional uncertainty sets for different constraints. In this case, the setting of Section 2, in which we consider in line with the common literature the same full-dimensional uncertainty set  $\mathcal{U}$  for all constraints, is not suitable.



Additionally, a general weakness of the common definition of the RRF in view of comparing the RRF for different models of the same problem is that scaling the constraints of the nominal problem (P) changes the RRF, which can make the RRF values meaningless in practice. We illustrate this by the following example.

**Example 3.1.** We consider the uncertainty set  $\mathcal{U} = [-1, 1]^2$ . Then, a nominal problem with constraints given by  $-x_1 \leq 1$ ,  $x_1 \in \mathbb{R}$ , has RRF 1 whereas scaling this nominal problem by a factor of 2 leads to  $-2x_1 \leq 2$ ,  $x_1 \in \mathbb{R}$ , with RRF 2.

The latter example and the above mentioned limitations of the setting for the RRF from Section 2 motivate us to extend this setting. This will allow us to apply the concept of the RRF to more MIP instances and applications such as computing the “most robust” solution in robust facility location design.

We now introduce our extended setting for the RRF of a MIP. In analogy to Section 2, we consider the nominal MIP (P). Let  $\alpha \geq 0$  be a fixed value and  $\mu^j$  the smallest absolute nonzero coefficient of the  $j$ th constraint of (P). The robust counterpart for the uncertain MIP (P) with uncertainty sets  $\alpha\bar{\mathcal{U}}_j, j \in J$ , is now given by

$$\min_{x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}} \{c^T x : (a^j)^T x \leq b^j, \forall (a^j, b^j) \in \{(\bar{a}^j, \bar{b}^j) + \alpha u : u \in \bar{\mathcal{U}}_j\}, j \in J\}, \text{ (EPR}_\alpha)$$

whereby for  $j \in J$  the uncertainty set  $\bar{\mathcal{U}}_j := \mu^j \mathcal{U}_j \subset \mathbb{R}^{n+1}$  is composed of a convex and compact set  $\mathcal{U}_j$  that is scaled by  $\mu^j$ . In contrast to (PR $_\alpha$ ) of Section 2, we now consider in (EPR $_\alpha$ ) for every constraint an own uncertainty set. These sets are not necessarily equal. Additionally, for  $j \in J$  every uncertainty set  $\bar{\mathcal{U}}_j$  is scaled by the smallest absolute nonzero coefficient of the  $j$ th constraint. The latter prevents that the RRF of a MIP can be increased by scaling the nominal problem such as in Example 3.1, which we will show later in this section, see Lemma 3.6. We note that the uncertain problem (PR $_\alpha$ ) of the previous section is a special case of the extended uncertain problem (EPR $_\alpha$ ).

In contrast to the setting of Section 2 that requires zero in the interior of the uncertainty set, see Assumption 1, we relax this condition such that zero is only a part of our uncertainty set. Consequently, the uncertainty set is not necessarily full-dimensional and we now can model safe variables and constraints. A variable  $x_i, i \in I$ , is said to be safe for the  $j$ th constraint,  $j \in J$ , if the projection of  $\bar{\mathcal{U}}_j$  on the  $i$ th axis equals  $\{0\}$ . Further, a variable  $x_i, i \in I$ , is said to be safe if it is safe for each constraint  $(\bar{a}^j)^T x \leq \bar{b}^j, j \in J$ .

For constraints, we now differentiate between two notions of being safe. A constraint  $(\bar{a}^j)^T x \leq \bar{b}^j, j \in J$ , is *syntactically* safe if  $\bar{\mathcal{U}}_j = \{0\}$ . It is *semantically* safe, if  $\delta^*((x, -1)^T \mid \bar{\mathcal{U}}_j) = 0$  for all feasible points  $x \in \mathbb{R}^n$  of (P). Whereas a syntactically safe constraint is also semantically safe, the converse statement is not necessarily true.

Considering the input data of the optimization problem, we can easily check whether a constraint is syntactically safe but not whether it is semantically safe. We note that decision makers can explicitly model syntactically safe constraints by setting the corresponding uncertainty set to zero. Throughout the following sections, we use safe as short form of semantically safe, if not explicitly stated otherwise.

The requirement that the uncertainty set contains zero is reasonable because it ensures that the nominal problem (EPR $_0$ ) is feasible for the RRF.

*Assumption 2.* Zero is contained in every uncertainty set  $\bar{\mathcal{U}}_j$  for  $j \in J$ .

In analogy to Section 2, we define the *radius of robust feasibility* (RRF) of a given MIP in our extended setting by

$$\rho_{\text{MIP}} := \sup\{\alpha \geq 0 : (\text{EPR}_\alpha) \text{ is feasible}\}.$$

Similar to Proposition 2.1, we reformulate the feasible region of the semi-infinite problem (EPR $_{\alpha}$ ) and obtain the ordinary robust counterpart

$$\min_{x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}} \{c^T x : (\bar{a}^j)^T x + \alpha \delta^*(x, -1)^T | \bar{U}_j \leq \bar{b}^j, j \in J\}. \quad (\text{EPRC}_{\alpha})$$

In analogy to Section 2, the robust counterparts corresponding to the continuous relaxation of (EPR $_{\alpha}$ ) equal the continuous relaxation of (EPRC $_{\alpha}$ ). We note that the setting of Section 2 is included in our extended setting of this section. We now compare the two settings and highlight similarities and differences.

First, we summarize all statements of Section 2 that are satisfied in our extended setting and can be shown analogously to the previous section.

*Observation 3.2.* Let  $\rho_{\text{MIP}}$  be the RRF of (P) and  $\rho_{\text{LP}}$  the RRF of its LP relaxation (LP). Then, the following statements hold:

- (i)  $0 \leq \rho_{\text{MIP}} \leq \rho_{\text{LP}}$ .
- (ii) MIPs with  $\rho_{\text{MIP}} < \rho_{\text{LP}}$  exist.
- (iii) MIPs exist such that the RRF  $\rho_{\text{LP}}$  is attained and  $\rho_{\text{MIP}}$  is attained.  
MIPs exist such that the RRF  $\rho_{\text{LP}}$  is attained and  $\rho_{\text{MIP}}$  is not attained.

We also note that Observation 2.3 and Lemma 2.9 are valid in our new setting, which can be shown in analogy to Section 2.

We now turn to the differences between the two considered settings for the RRF. Using counterexamples we show that several statements of the previous Section 2 are not satisfied in our extended setting. Especially, the main result, Statement (i) of Theorem 2.6, is not satisfied anymore. First, we note that the RRF is not necessarily finite in our new setting, which especially holds for every feasible nominal problem (P) if the uncertainty set contains only zero.

*Observation 3.3.* MIPs exist such that the RRF of (P) is infinite.

The next example shows that if the RRF  $\rho_{\text{LP}}$  of the LP relaxation (LP) is not attained, then the RRF  $\rho_{\text{MIP}}$  of (P) is not necessarily equal to  $\rho_{\text{LP}}$ .

**Example 3.4.** The constraints of the nominal problem are given by

$$x_1 \leq 1, \quad -x_1 \leq 0.1, \quad -x_2 \leq -2, \quad x_1 \in \mathbb{Z}, x_2 \in \mathbb{R}, \quad (10)$$

with the uncertainty sets  $\bar{U}_1 = [0]^2 \times [-0.5, 0.5]$ ,  $\bar{U}_2 = 0.1 \cdot ([0]^2 \times [-5, 5])$ , and  $\bar{U}_3 = [0] \times [-1, 1] \times [0]$ . Proposition 2.1 leads to the robust counterpart of (10)

$$x_1 \leq 1 - 0.5\alpha, \quad -x_1 \leq 0.1 - 0.5\alpha, \quad -x_2 + \alpha|x_2| \leq -2, \quad x_1 \in \mathbb{Z}, x_2 \in \mathbb{R}. \quad (11)$$

From Counterpart (11) it follows that the RRF  $\rho_{\text{MIP}}$  of (10) equals 0.2 and it is attained by any point  $(0, x_2)$  such that  $x_2 \geq 2.5$ .

We now consider the LP relaxation of (10) and the corresponding counterpart, which is the continuous relaxation of (11). For every  $\alpha \in [0, 1)$  the element  $(x_1, x_2) = (0.5, 2/(1 - \alpha))$  is feasible for the continuous robust counterpart. Furthermore, for  $\alpha = 1$  the corresponding counterpart is infeasible because  $-x_2 + |x_2| \leq -2$ ,  $x_2 \in \mathbb{R}$ , cannot be satisfied. Consequently, the RRF  $\rho_{\text{LP}}$  of the LP relaxation of (10) equals 1 and is not attained by a feasible solution.

From Example 3.4 it follows that the main result of Section 2, Statement (i) of Theorem 2.6, is not valid in our new setting for the RRF. Furthermore, Statement (ii) of Theorem 2.6 does not hold.

**Lemma 3.5.** Let  $\rho_{\text{MIP}}$  be the RRF of (P) and  $\rho_{\text{LP}}$  the RRF of its LP relaxation (LP). Then, the following statements hold:

- (i) MIPs exist such that  $\rho_{\text{LP}}$  is not attained and  $\rho_{\text{MIP}} < \rho_{\text{LP}}$  holds.
- (ii) MIPs exist such that the RRF  $\rho_{\text{MIP}}$  is attained and  $\rho_{\text{LP}}$  is not attained.

In addition that we can now handle safe constraints and variables in our extended setting, we now prove that scaling the nominal problem by a positive factor does not change the RRF, which is not valid for the RRF in the setting of Section 2, see Example 3.1.

**Lemma 3.6.** *Let  $\rho_{\text{MIP}}$  be the RRF of (P) and  $\lambda^j > 0, j \in J$ , positive factors. Then,  $\rho_{\text{MIP}}$  is also the RRF of the  $\lambda$ -scaled problem (P), i.e.,  $\rho_{\text{MIP}}$  is the RRF of*

$$\min_{x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}} \{c^T x : \lambda^j (\bar{a}^j)^T x \leq \lambda^j \bar{b}^j, j \in J\}. \quad (12)$$

*Proof.* For every  $j \in J$ , scaling the  $j$ th constraint of the nominal problem (P) by  $\lambda^j$  also scales the smallest absolute nonzero coefficient  $\mu^j$  of the  $j$ th constraint by  $\lambda^j$ . Hence, for  $j \in J$  the uncertainty set  $\bar{U}_j$  is scaled by  $\lambda^j$ . From this it follows that the uncertain problem of (12) equals (EPR $_{\alpha}$ ). Thus,  $\rho_{\text{MIP}}$  is also the RRF of (12).  $\square$

We now have analyzed similarities and differences for the setting of the RRF in Section 2 and our extended setting. To conclude this section, we present a necessary optimality condition for the RRF of a MIP in our extended setting that we then extend to a necessary and sufficient condition under additional assumptions. Its basic idea is rather simple, if none of the constraints is tight for a considered feasible solution, then we can increase the uncertainty set which implies that the chosen size of the uncertainty set was not maximal.

**Theorem 3.7.** *Let  $\alpha \geq 0$  be the finite RRF of (P). Then, for every feasible solution  $x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}$  of (EPR $_{\alpha}$ ) there exists an index  $j \in J$  which satisfies*

$$(\bar{a}^j)^T x + \alpha \delta^*((x, -1)^T | \bar{U}_j) = \bar{b}^j.$$

*Proof.* Let  $\alpha \in \mathbb{R}$  be the finite RRF of (P). We contrarily assume that  $(\varepsilon, x)$  with  $\varepsilon > 0$  exists such that  $x$  is feasible for (EPR $_{\alpha}$ ) and

$$(\bar{a}^j)^T x + \alpha \delta^*((x, -1)^T | \bar{U}_j) + \varepsilon \leq \bar{b}^j, \quad j \in J \setminus S, \quad (13)$$

is satisfied, whereby  $\delta^*((x, -1)^T | \bar{U}_j)$  is positive only for  $j \in J \setminus S$ . We note that  $J \setminus S$  is nonempty, because the RRF is finite. Further, the support function  $\delta^*((x, -1)^T | \bar{U}_j)$  is nonnegative because for  $j \in J$  the uncertainty set  $\bar{U}_j$  contains zero. The inequalities

$$\alpha \leq \frac{\bar{b}^j - (\bar{a}^j)^T x - \varepsilon}{\delta^*((x, -1)^T | \bar{U}_j)}, \quad j \in J \setminus S$$

hold, which follows from (13). We now set

$$\alpha' = \min_{l \in J \setminus S} \frac{\bar{b}^l - (\bar{a}^l)^T x}{\delta^*((x, -1)^T | \bar{U}_l)}.$$

Then,  $\alpha' > \alpha$  holds because  $\varepsilon$  is positive. Furthermore, for  $j \in J \setminus S$  the inequality

$$(\bar{a}^j)^T x + \alpha' \delta^*((x, -1)^T | \bar{U}_j) \leq (\bar{a}^j)^T x + \frac{\bar{b}^j - (\bar{a}^j)^T x}{\delta^*((x, -1)^T | \bar{U}_j)} \delta^*((x, -1)^T | \bar{U}_j) \leq \bar{b}^j$$

is satisfied. Consequently, the solution  $x$  is feasible for (EPR $_{\alpha'}$ ). This shows together with Observation 2.3 that  $\alpha$  cannot be the RRF of (P).  $\square$

In the following, the index set  $S_{\text{MIP}} \subseteq J$  contains all ‘‘safe’’ constraints, i.e., for every feasible solution  $x$  of (P) the equality  $\delta^*((x, -1)^T | \bar{U}_j) = 0$  holds for  $j \in S_{\text{MIP}}$ . If the RRF of a given MIP is attained and for each feasible solution  $x$  of (P) the counterpart  $\delta^*((x, -1)^T | \bar{U}_j)$  is positive for  $j \in J \setminus S_{\text{MIP}}$ , then the

previous necessary optimality condition can be extended to a necessary and sufficient optimality condition. To this end, we introduce the optimization problem  $(\text{SPRC}_\alpha)$

$$\begin{aligned} & \sup_{x, \varepsilon} \quad \varepsilon \\ & \text{s.t.} \quad (\bar{a}^j)^T x + \alpha \delta^*((x, -1)^T \mid \bar{U}_j) + \varepsilon \leq \bar{b}^j, \quad j \in J \setminus S_{\text{MIP}}, \quad (\text{SPRC}_\alpha) \\ & \quad (\bar{a}^j)^T x \leq \bar{b}^j, \quad j \in S_{\text{MIP}}, \quad x \in \mathbb{Z}^k \times \mathbb{R}^{n-k}, \quad \varepsilon \geq 0. \end{aligned}$$

**Lemma 3.8.** *Let  $\rho_{\text{MIP}}$  be the RRF of (P),  $\alpha \geq 0$ , and for every feasible solution  $x$  of (P) the inequality  $\delta^*((x, -1)^T \mid \bar{U}_j) > 0$  holds for  $j \in J \setminus S_{\text{MIP}}$ . If the optimal objective value of  $(\text{SPRC}_\alpha)$  is zero, then it is attained and  $\alpha$  equals  $\rho_{\text{MIP}}$ .*

*Proof.* Due to the optimal objective value being zero and constraint  $\varepsilon \geq 0$ , Problem  $(\text{SPRC}_\alpha)$  is feasible and every feasible solution  $(\varepsilon, x)$  satisfies  $\varepsilon = 0$ . Consequently, the optimal objective value is attained. For a given  $\alpha \geq 0$ , let  $(0, x)$  be an optimal solution of  $(\text{SPRC}_\alpha)$ . We now assume that  $\alpha \neq \rho_{\text{MIP}}$  holds. If  $\alpha > \rho_{\text{MIP}}$  is satisfied, then this is a contradiction to the optimality of the RRF  $\rho_{\text{MIP}}$  due to Observation 2.3 and the feasibility of  $(\text{SPRC}_\alpha)$ . We now assume  $0 \leq \alpha < \rho_{\text{MIP}}$ . Consequently,  $\alpha'$  with  $0 \leq \alpha < \alpha' \leq \rho_{\text{MIP}}$  and a solution  $x'$  exists such that

$$(\bar{a}^j)^T x' + \alpha' \delta^*((x', -1)^T \mid \bar{U}_j) \leq \bar{b}^j, \quad j \in J, \quad (14)$$

holds. Due to the requirements  $\delta^*((x', -1)^T \mid \bar{U}_j) > 0$  for  $j \in J \setminus S_{\text{MIP}}$  is satisfied and thus, from (14) follows

$$(\bar{a}^j)^T x' + \alpha \delta^*((x', -1)^T \mid \bar{U}_j) < \bar{b}^j, \quad j \in J \setminus S_{\text{MIP}}.$$

Consequently, the objective value of  $(\text{SPRC}_\alpha)$  is

$$\varepsilon = \min_{j \in J \setminus S_{\text{MIP}}} \bar{b}^j - (\bar{a}^j)^T x' - \alpha \delta^*((x', -1)^T \mid \bar{U}_j) > 0$$

for  $(\varepsilon, x')$ . This is a contradiction to the optimality of  $(0, x)$  for  $\alpha$ . Thus,  $\alpha = \rho_{\text{MIP}}$  is satisfied.  $\square$

Finally, we present our necessary and sufficient optimality condition for the RRF.

**Theorem 3.9.** *Let the RRF  $\rho_{\text{MIP}}$  of (P) be attained,  $S_{\text{MIP}} \neq J$ , and for every feasible solution  $x$  of (P) the inequality  $\delta^*((x, -1)^T \mid \bar{U}_j) > 0$  holds for  $j \in J \setminus S_{\text{MIP}}$ . Then, the value  $\alpha$  equals  $\rho_{\text{MIP}}$  if and only if the optimal objective value of  $(\text{SPRC}_\alpha)$  equals zero.*

*Proof.* The RRF is attained, i.e.,  $(\text{EPRC}_{\rho_{\text{MIP}}})$  is feasible. Moreover, for every feasible solution  $x$  of (P) the inequality  $\delta^*((x, -1)^T \mid \bar{U}_j) > 0$  holds for  $j \in J \setminus S_{\text{MIP}}$  with  $S_{\text{MIP}} \neq J$  and thus, the RRF is finite. Let  $\alpha$  be equal to the RRF  $\rho_{\text{MIP}}$  and  $\varepsilon$  the optimal objective value of  $(\text{SPRC}_\alpha)$ . Since the RRF is attained,  $(\text{EPR}_\alpha)$  and  $(\text{SPRC}_\alpha)$  are feasible. Consequently,  $\varepsilon$  cannot equal zero while being not attained. If  $\varepsilon$  is positive, then a feasible solution  $(\varepsilon, x)$  of  $(\text{SPRC}_\alpha)$  with  $\varepsilon > 0$  exists. This is a contradiction to the optimality of  $\alpha$  because of Theorem 3.7 and its proof. Consequently,  $\varepsilon$  equals zero and is attained by a feasible solution of  $(\text{SPRC}_\alpha)$ . Thus, the claim is shown by Lemma 3.8.  $\square$

Theorem 3.9 is valid for the setting of Section 2 without assuming  $\delta^*((x, -1)^T \mid \mathcal{U}) > 0$  for every feasible solution  $x$  of (P) because Assumption 1 implies the latter.

We now move on to the computation of the RRF for LPs as well as MIPs including safe variables and constraints in our extended setting.

#### 4. COMPUTING THE RRF INCLUDING SAFE CONSTRAINTS AND VARIABLES

Many known techniques for computing the RRF rely on full-dimensional uncertainty sets and compute the RRF for continuous problems, see [18, 19, 22, 23, 29]. Hence, it is not obvious if and how these techniques can be applied to our extended setting of Section 3 in which MIPs with different not necessarily full-dimensional uncertainty sets are considered. The latter enables us to consider MIPs including safe variables and constraints. Consequently, there is a lack of methods that compute the RRF for LPs as well as for MIPs including safe variables and constraints. This section is structured as follows. We first show a method for computing the RRF of LPs including safe variables and constraints. We then briefly show that the RRF of a bounded integer problem can be computed by solving maximally two integer problems. Finally, we present first methods for computing the RRF of MIPs in our extended setting of Section 3.

**4.1. Computing the RRF for Linear Problems.** In this subsection, we present a method for computing the RRF of (LP). To this end, we consider our general setting of Section 3. Throughout this section, we split the constraints of (LP) into “safe” constraints  $S_{LP} \subseteq J$ , i.e. for every feasible solution  $x$  of (LP) the equality  $\delta^*((x, -1) \mid \bar{U}_j) = 0$  holds for  $j \in S_{LP}$ , and into “uncertain” constraints  $J \setminus S_{LP}$ . Additionally, we require the following assumption for the uncertainty sets.

*Assumption 3.* We assume for the uncertain constraints that, up to scaling, all uncertainty sets are identical, i.e.,  $\bar{U}_j = \mu^j \lambda^j \mathcal{U} \subset \mathbb{R}^{n+1}$  for  $j \in J \setminus S_{LP}$  holds whereby  $\mathcal{U}$  is a convex and compact uncertainty set and  $\lambda^j$  is positive for  $j \in J \setminus S_{LP}$ .

We note that typically the uncertainty sets  $\bar{U}_j, j \in J$ , are positive multiples of the Euclidean unit closed ball or of some cartesian product  $\mathcal{U} = \prod_{i \in I} [-\delta_i, \delta_i]$ , with  $\delta_i \geq 0$  for all  $i \in I$ , which is in line with Assumption 3. Moreover, the positive homogeneity of  $\delta^*(x \mid \bar{U}_j)$  for  $j \in J$  and Assumption 3 lead to

$$\delta^*((x, -1)^T \mid \bar{U}_j) = \mu^j \lambda^j \delta^*((x, -1)^T \mid \mathcal{U}), \quad j \in J \setminus S_{LP}. \quad (15)$$

Thus, under Assumption 3, for all feasible points  $x \in \mathbb{R}^n$  of (LP), the equality  $\delta^*((x, -1)^T \mid \bar{U}_j) = 0$  either holds for all  $j \in J \setminus S_{LP}$  or for no index  $j \in J \setminus S_{LP}$ .

We note that this setting is more general than that of Section 2 because it does not require a full-dimensional uncertainty set and thus, we allow safe variables and constraints. We further consider objective functions as extended-value functions whereby we follow the extended-value definition in [27]. Consequently, if an optimization problem is infeasible, then its objective value is  $+\infty$  for minimization problems, respectively  $-\infty$  for maximization problems. Furthermore,  $\frac{1}{+\infty} := 0$  and  $\frac{1}{0} := +\infty$  hold.

We now give a derivation of our method that is based on fractional programming. We first handle the case that a feasible solution  $x$  of (LP) without uncertainty exists, i.e.,  $\delta^*((x, -1)^T \mid \mathcal{U}) = 0$ . Then, we consider the case that the RRF is zero. Afterward, we present a method that computes the RRF if the latter is positive. Finally, we combine these results in an algorithm that computes the RRF for LPs.

Clearly, if a feasible solution of (LP) which is not affected by any uncertainty exists, then the RRF is infinite.

**Proposition 4.1.** *Let  $x \in \mathbb{R}^n$  be a feasible solution to (LP) such that the equality  $\delta^*((x, -1)^T \mid \mathcal{U}) = 0$  holds. Then, the RRF of (LP) is infinite.*

Next, we show that the requirement of Proposition 4.1 can be checked algorithmically. We know that  $\delta^*((\cdot, -1)^T \mid \mathcal{U}) \geq 0$  holds due to Assumption 2. Consequently, we can verify if the equation  $\delta^*((x, -1)^T \mid \mathcal{U}) = 0$  holds for any feasible solution  $x$

of (LP) by checking the feasibility of the convex problem

$$\min_x 0 \quad (16a)$$

$$\text{s.t. } (\bar{a}^j)^T x \leq \bar{b}^j \quad \text{for all } j \in J, \quad (16b)$$

$$\delta^*((x, -1)^T | \mathcal{U}) \leq 0. \quad (16c)$$

**Lemma 4.2.** *A feasible solution  $x$  of (LP) with  $\delta^*((x, -1)^T | \mathcal{U}) = 0$  exists if and only if Problem (16) is feasible.*

We now assume that for every feasible solution  $x$  of the nominal problem (LP) the inequality  $\delta^*((x, -1)^T | \mathcal{U}) > 0$  holds, since otherwise the RRF is infinite which we can detect by the previously stated model.

Due to the definition of the RRF, the feasibility of (LP), and Assumption 3, the RRF of Problem (LP) can be computed by the nonlinear problem

$$\begin{aligned} \sup_{\alpha, x} \quad & \alpha \\ \text{s.t.} \quad & (\bar{a}^j)^T x + \alpha \mu^j \lambda^j \delta^*((x, -1)^T | \mathcal{U}) \leq \bar{b}^j \quad \text{for all } j \in J \setminus S_{\text{LP}}, \\ & (\bar{a}^j)^T x \leq \bar{b}^j \quad \text{for all } j \in S_{\text{LP}}, \end{aligned}$$

which we can reformulate as

$$\sup_x \min_{j \in J \setminus S_{\text{LP}}} \frac{\bar{b}^j - (\bar{a}^j)^T x}{\mu^j \lambda^j \delta^*((x, -1)^T | \mathcal{U})} \quad (17a)$$

$$\text{s.t. } (\bar{a}^j)^T x \leq \bar{b}^j \quad \text{for all } j \in S_{\text{LP}}. \quad (17b)$$

Problem (17) is a generalized fractional program. Additionally, for every feasible solution of the nominal problem (LP) and for every ratio in the objective function the corresponding nominator is nonnegative and concave and the denominator is positive and convex. Thus, Problem (17) has the form of a concave generalized fractional program, see [2, Chapter 7]. We now reduce Problem (17) to a concave single ratio fractional program, which then can be reformulated as a concave problem. To this end, we reformulate Problem (17) as follows

$$\sup_{x, \varepsilon, z} \frac{z}{\delta^*((x, -1)^T | \mathcal{U})} \quad (18a)$$

$$\text{s.t. } (\bar{a}^j)^T x + \varepsilon_j \leq \bar{b}^j \quad \text{for all } j \in J \setminus S_{\text{LP}}, \quad (18b)$$

$$(\bar{a}^j)^T x \leq \bar{b}^j \quad \text{for all } j \in S_{\text{LP}}, \quad (18c)$$

$$\varepsilon_j \geq 0 \quad \text{for all } j \in J \setminus S_{\text{LP}}, \quad (18d)$$

$$\frac{\varepsilon_j}{\mu^j \lambda^j} \geq z \geq 0 \quad \text{for all } j \in J \setminus S_{\text{LP}}. \quad (18e)$$

We note that Problem (18) is a concave fractional program with a single ratio in the objective function. Furthermore, the RRF of (LP) is strictly positive if and only if a feasible solution  $(x, \varepsilon, z)$  of (18) with  $z > 0$  exists because  $\delta^*((x, -1)^T | \mathcal{U}) > 0$  holds. We now assume that the variable  $z$  is positive and show that we can algorithmically check if the RRF is zero with the help of a linear problem.

$$\sup_{x, \varepsilon, z} \frac{z}{\delta^*((x, -1)^T | \mathcal{U})} \quad \text{s.t. } (18b) - (18e), \quad z > 0. \quad (19)$$

**Lemma 4.3.** *Problem (19) is feasible if and only if the RRF of (LP) is strictly positive.*

*Proof.* Let  $(x, \varepsilon, z)$  satisfy constraints (18b)–(18e). Then,  $(x, \varepsilon, z)$  is feasible for Problem (19) if and only if  $z > 0$  holds, which in turn is equivalent to the optimal value of Problem (18) being strictly positive.  $\square$

Clearly, we can check if Problem (19) is feasible by solving a linear problem.

**Lemma 4.4.** *Problem (19) is feasible if and only if the objective value of problem*

$$\max_{x,\varepsilon,z} z \quad \text{s.t.} \quad (18b) - (18e) \quad (20)$$

*is positive.*

Using linear Problem (20), we can detect whether the RRF of (LP) is strictly positive or zero. We now handle the case that the RRF is strictly positive. Thus, we consider the optimization problem, in which we minimize the reciprocal of the original objective function of (19)

$$\inf_{x,\varepsilon,z} \frac{\delta^*((x, -1)^T \mid \mathcal{U})}{z} \quad \text{s.t.} \quad (18b) - (18e), \quad z > 0. \quad (21)$$

We note that Problem (19) and (21) have the same feasible region and for every feasible solution the corresponding objective value is positive. Both problems share the same optimal solutions and the optimal values are reciprocal to each other. Throughout this section, we consider objective values in the extended-value sense.

**Lemma 4.5.** *Let  $(x, \varepsilon, z)$  be a feasible solution of Problem (19). Then,  $(x, \varepsilon, z)$  is an optimal solution of Problem (19), if and only if  $(x, \varepsilon, z)$  is an optimal solution of Problem (21).*

*Let  $v$  and  $\hat{v}$  be the optimal values of (19) and (21). Then, the equation  $v = \frac{1}{\hat{v}}$  holds in the extended-value sense.*

Due to Lemma 4.5, the optimal value of (21) is zero if and only if the RRF of (LP) is infinite. Problem (21) is equivalent to a concave fractional program with affine denominator. Thus, we can apply a variable transformation that was suggested by Charnes and Cooper [17] for linear fractional programs and later extended to nonlinear fractional programs by Schaible [31], see also [2] and the references therein. The transformation is given by

$$y = \begin{bmatrix} y_x \\ y_\varepsilon \\ y_z \end{bmatrix} = \frac{1}{z} \begin{bmatrix} x \\ \varepsilon \\ z \end{bmatrix}, \quad t = \frac{1}{z}. \quad (22)$$

Applying this variable transformation to Problem (21) together with Proposition 7.2 in Chapter 7 of [2] and the positive homogeneity of the support function lead us to the following lemma.

**Lemma 4.6.** *Let  $(x, \varepsilon, z)$  and  $(y, t)$  be given such that Transformation (22) holds. Then  $(x, \varepsilon, z)$  is feasible for Problem (21) if and only if  $(y, t)$  is feasible to problem*

$$\inf_{y,t} \delta^*((y_x, -t)^T \mid \mathcal{U}) \quad (23a)$$

$$\text{s.t.} \quad (\bar{a}^j)^T y_x + y_{\varepsilon_j} - t\bar{b}^j \leq 0 \quad \text{for all } j \in J \setminus S_{LP}, \quad (23b)$$

$$(\bar{a}^j)^T y_x - t\bar{b}^j \leq 0 \quad \text{for all } j \in S_{LP}, \quad (23c)$$

$$y_{\varepsilon_j} \geq \mu^j \lambda^j \quad \text{for all } j \in J \setminus S_{LP}, \quad (23d)$$

$$t > 0. \quad (23e)$$

*Furthermore, the optimal values of Problems (21) and (23) are equal.*

We now relax Problem (23) by requiring  $t \geq 0$  instead of  $t > 0$  in order to obtain the computationally tractable convex optimization problem

$$\inf_{y,t} \delta^*((y_x, -t)^T \mid \mathcal{U}) \quad \text{s.t.} \quad (23b) - (23d), \quad t \geq 0. \quad (24)$$

All constraints of Problem (24) are linear. Furthermore, optimizing (24) has the same computational complexity as optimizing a linear function over the given uncertainty set  $\mathcal{U}$  with additional linear constraints.

We note that if Problem (23) is feasible, then the objective values of (23) and (24) are finite due to Assumption 2, which implies  $\delta^*((y_x, -t)^T | \mathcal{U}) \geq 0$ . Further, we now prove that these objective values are equal.

**Lemma 4.7.** *Let Problem (23) be feasible. Then, the optimal values of Problems (23) and (24) are equal.*

*Proof.* Let  $v$  be the optimal value of Problem (23) and  $v_{\text{relax}}$  the optimal value of Problem (24). As Problem (24) is a relaxation of Problem (23),  $v_{\text{relax}} \leq v$  holds.

So, assume, by contradiction, that there exists a solution  $(y^*, t^*)$  of Problem (24) with corresponding objective value  $v^*$  and  $v^* < v$ . Thus,  $t^* = 0$  holds. As Problem (23) is feasible, there exists a feasible point  $(\bar{y}, \bar{t})$  with  $\bar{t} > 0$ .

Now, set

$$\begin{bmatrix} y^k \\ t^k \end{bmatrix} = \frac{k-1}{k} \begin{bmatrix} y^* \\ t^* \end{bmatrix} + \frac{1}{k} \begin{bmatrix} \bar{y} \\ \bar{t} \end{bmatrix} \quad \text{for all } k \in \mathbb{N}.$$

Then, the pairs  $(y^k, t^k)$  are feasible for Problem (24) as its feasible region is convex. Since  $t^k > 0$  holds,  $(y^k, t^k)$  is also feasible to (23). The objective values  $v^k$  of these solutions converge to  $v^*$  as the support function,  $\delta^*((y_x^k, -t^k)^T | \mathcal{U})$ , is continuous. Hence, there exists a  $\bar{k} \in \mathbb{N}$  such that  $v^{\bar{k}} < v$  holds, which contradicts the fact that  $v$  is the optimal value of Problem (23). Consequently, the optimal values of Problem (24) and of Problem (23) are equal.  $\square$

Again, we use extended-values in this section.

**Lemma 4.8.** *Let Problem (23) be feasible and  $v$  the optimal value of (24). Then, the RRF of (LP) is given by  $\frac{1}{v}$ .*

*Proof.* The claim follows from combining the previous Lemmas 4.7, 4.6, and 4.5.  $\square$

Using the previous results, we now state a complete procedure to compute the RRF of (LP) whereby the uncertainty sets satisfy Assumption 2 and 3.

---

**Algorithm 1:** Computing the RRF of a Linear Problem

---

**Input:** Linear Problem (LP) and uncertainty sets  $\bar{\mathcal{U}}_j$  for  $j \in J$ .

**Output:** RRF of (LP).

- 1 **if** Problem (16) is feasible **then return**  $+\infty$ .
  - 2 Solve  $(x, \varepsilon, z) \leftarrow$  (20).
  - 3 **if**  $z = 0$  holds **then return** 0.
  - 4 Compute optimal objective  $v$  of Problem (24).
  - 5 **return**  $\frac{1}{v}$ .
- 

**Theorem 4.9.** *Let the robust counterpart for the uncertain linear problem (LP) with uncertainty sets  $\alpha\bar{\mathcal{U}}_j, j \in J$ , satisfy Assumptions 2 and 3. Then, Algorithm 1 computes the RRF of (LP).*

*Proof.* If Algorithm 1 stops in Line 1, then the RRF is infinite due to Proposition 4.1 and Lemma 4.2. If Algorithm 1 stops in Line 3, then the RRF is zero due to Lemmas 4.3 and 4.4. If Algorithm 1 stops in Line 5, then the feasibility of (23) follows from the positive objective value of (20) and Lemmas 4.4–4.6. Thus, we can apply Lemma 4.8, which proves the claim.  $\square$



In summary, we can efficiently compute the RRF of (LP) including safe variables and constraints by solving at most one linear and two convex optimization problems. Especially, solving the latter problems has the same computational complexity as optimizing a linear objective function over the given uncertainty set with additional linear constraints. In addition to the benefit for computing the RRF of LPs, the results can be used as an upper bound for the RRF of the corresponding MIPs which will be helpful for the later presented methods. We also note that under certain conditions the RRF of a MIP can be computed by the RRF of its LP relaxation, see Theorem 2.6.

**4.2. Computing the RRF of Bounded Integer Problems.** In this subsection, we briefly show that for a bounded linear integer problem we can compute its RRF in the setting of Section 3 by maximally solving two convex integer problems. The latter problems have the same complexity as solving an integer problem with linear objective function over the given uncertainty set with additional linear constraints. For the remainder of this subsection, we assume w.l.o.g. that our bounded integer problem (P) is a binary problem.

We first show that the compactness of the feasible region of (P) ensures that its RRF is either attained or infinite and the latter can be checked algorithmically. To this end, we note that Lemma 2.9 is also valid for a finite RRF in the setting of Section 3 which can be proven analogously.

**Lemma 4.10.** *If the feasible region of (P) is compact, then the corresponding RRF is either attained or infinite.*

*Proof.* We contrarily assume that the RRF is not attained and finite. Due to Lemma 2.9 an unbounded sequence of feasible solutions to (P) exists, which contradicts the compactness of the feasible region of (P).  $\square$

Additionally, we can detect if the RRF is infinite. In doing so, the index set  $S_{MIP}$  contains all safe constraints of (P) that are not affected by uncertainty.

**Lemma 4.11.** *Let the feasible region of (P) be compact. Then, the RRF of (P) is infinite if and only if the convex integer problem*

$$\min_x 0 \quad \text{s.t.} \quad (16b), \quad \delta^*((x, -1)^T | \bar{U}_j) \leq 0, j \in J \setminus S_{MIP}, \quad x \in \{0, 1\}^n, \quad (25)$$

*is feasible.*

*Proof.* We first assume that the RRF of (P) is infinite. Due to the requirements and the definition of the RRF, a positive and strictly increasing sequence  $(\alpha^l)_{l \in \mathbb{N}}$  that converges to  $+\infty$  exists. Furthermore, a sequence in  $\mathbb{R}^n$ ,  $(x^l)_{l \in \mathbb{N}}$ , exists such that  $x^l$  is feasible to  $(EPRC_{\alpha^l})$  for all  $l \in \mathbb{N}$ . Due to the compactness of the feasible region of (P), the sequence  $(x^l)_{l \in \mathbb{N}}$  is bounded. Consequently, and by passing to a subsequence if necessary, we may assume that  $x^l \rightarrow \bar{x}$  holds. Considering  $(EPRC_{\alpha^l})$  together with a solution  $x^l$  leads to the feasible inequalities

$$(\bar{a}^j)^T x^l + \alpha^l \delta^*((x^l, -1)^T | \bar{U}_j) \leq \bar{b}^j, \quad j \in J \setminus S_{MIP}.$$

Since sequence  $(x^l)_{l \in \mathbb{N}}$  is bounded,  $(\alpha^l)_{l \in \mathbb{N}}$  converges to  $+\infty$ , and  $\delta^*((x^l, -1)^T | \bar{U}_j)$  is nonnegative for  $j \in J \setminus S_{MIP}$ , it follows from the previous inequalities that for  $j \in J \setminus S_{MIP}$  the support function  $\delta^*((x^l, -1)^T | \bar{U}_j)$  converges to zero. Due to this,  $x^l \rightarrow \bar{x}$ , and the continuity of  $\delta^*((x^l, -1)^T | \bar{U}_j)$  for  $j \in J \setminus S_{MIP}$ , the equality  $\delta^*((\bar{x}, -1)^T | \bar{U}_j) = 0$  holds for  $j \in J \setminus S_{MIP}$ . Because of the compactness of the feasible region and  $x^l \rightarrow \bar{x}$ , the solution  $\bar{x}$  is feasible to (P). Thus, it is feasible to (25).

If Problem (25) is feasible, then from the nonnegativity of the support function  $\delta^*((\cdot, -1)^T | \bar{U}_j)$  for  $j \in J \setminus S_{MIP}$ , it directly follows that the RRF is infinite.  $\square$

Due to the previous two lemmas, we can algorithmically check if the RRF of (P) is infinite. Thus, we now assume that the RRF is finite. Consequently, the RRF is attained because of Lemma 4.10 and we can compute the RRF by solving the nonlinear problem

$$\max_{\alpha, x} \alpha \quad (26a)$$

$$\text{s.t. } (\bar{a}^j)^T x + \delta^*((\alpha x, -\alpha)^T \mid \bar{U}_j) \leq \bar{b}^j \quad \text{for all } j \in J \setminus S_{\text{MIP}}, \quad (26b)$$

$$(\bar{a}^j)^T x \leq \bar{b}^j \quad \text{for all } j \in S_{\text{MIP}}, \quad (26c)$$

$$\alpha \geq 0, x \in \{0, 1\}^n. \quad (26d)$$

We can equivalently replace the nonlinear term  $\alpha x$  in (26) by suitable Big-M constraints because the RRF of (P) is finite and  $x$  are binaries.

In summary, we can compute the RRF for bounded integer problems in the setting of Section 3 by solving (25) and (26). This method is straightforward and is mainly presented for the sake of completeness. Furthermore, preliminary computational results showed that its performance is bad in general and cannot be used for practical computations. It is also massively worse in comparison to the methods of the next section that are based on improved effective binary search algorithms.

**4.3. Computing the RRF of Mixed-Integer Problems.** In this subsection, we present different methods to compute the RRF of MIPs (P) in our extended setting of Section 3 in the case that the RRF is finite. In doing so, the presented methods share a common basic structure, see Algorithm 2. We note that the considered setting of the RRF includes safe constraints, respectively variables, and an own not necessarily full-dimensional uncertainty set for every constraint.

For the remainder of this subsection, we assume that the RRF of (P) is finite and bounded from above by  $\bar{u}$ . Further, we know that the RRF is bounded from below by zero. In analogy to Observation 2.3 for  $\alpha \geq 0$  a monotonicity statement w.r.t. the corresponding ordinary counterpart (EPRC $_{\alpha}$ ) holds. Thus, we can apply a classic binary search (ClassicBin) on  $\alpha$  w.r.t. (EPRC $_{\alpha}$ ) in order to find an approximation of the RRF. This approximation differs from the RRF no more than an a priori given error  $\text{tol} > 0$ . Binary search is already in itself an efficient algorithm. However, we show in addition that our theoretical findings on RRF can be used to even improve on binary search in practical computations.

**Lemma 4.12.** *Let  $\rho_{\text{MIP}}$  be the RRF of (P). Further, let  $\alpha$  be the output of ClassicBin with initial lower bound zero,  $\bar{u}$  an upper bound of the RRF, and the tolerance  $\text{tol}$ . Then, (EPRC $_{\alpha}$ ) is feasible,  $|\rho_{\text{MIP}} - \alpha| \leq \text{tol}$  holds, and ClassicBin performs at most  $\lceil \log_2(\frac{\bar{u}}{\text{tol}}) \rceil$  many iterations.*

An important benefit of this simple approach is that in each step of the binary search it is sufficient to only check the feasibility of (EPRC $_{\alpha}$ ). With the help of standard techniques of robust optimization, e.g., see [4], Problem (EPRC $_{\alpha}$ ) can be reformulated such that its computational complexity is equal to checking the feasibility of an optimization problem over the given uncertainty set with additional linear constraints.

We now improve ClassicBin by adding a scaling argument so that whenever (EPRC $_{\alpha}$ ) is feasible, we tighten the lower bound in the binary search. To this end, Algorithm 2 represents the basic structure of this scaling binary search (ScalingBin) and its explicit components are given in Table 1. Method ScalingBin still maintains the properties of ClassicBin.

**Lemma 4.13.** *Let  $\rho_{\text{MIP}}$  be the RRF of (P). Further, let  $\alpha$  be the output of ScalingBin. Then, (EPRC $_{\alpha}$ ) is feasible,  $|\rho_{\text{MIP}} - \alpha| \leq \text{tol}$  holds, and ScalingBin performs at most  $\lceil \log_2(\frac{\bar{u}}{\text{tol}}) \rceil$  many iterations.*

**Algorithm 2:** Basic Algorithm

**Input:** Nominal problem (P), uncertainty sets  $\bar{U}_j$  for  $j \in J$ , tolerance  $\text{tol} > 0$ , RRF upper bound  $\bar{u}$ .

**Output:** RRF of (P).

1	Initialization.	Init
2	<b>while</b> Condition <b>do</b>	
3	Update Estimate RRF.	Estim
4	Solve Subproblem.	Subp
5	Check Optimality.	Optim
6	Update Upper Bound.	Upper
7	Update Lower Bound.	Lower
8	<b>return</b> Results.	

TABLE 1. Overview of algorithms with their specific components in Algorithm 2

	ScalingBin	MaxScalingBin	PureScaling
Init	$l \leftarrow 0, u \leftarrow \bar{u}$		$l \leftarrow 0$
Condition	$ u - l  > \text{tol}$		(EPRC $_{(l+\text{tol})}$ ) feasible
Estim	$\alpha \leftarrow \frac{u+l}{2}$		$\alpha \leftarrow l + \text{tol}$
Subp	$x \leftarrow (\text{EPRC}_\alpha)$	$(\varepsilon, x) \leftarrow (\text{SPRC}_\alpha)$	
Optim	if $\varepsilon = 0$ then return $(\alpha, \text{optimal})$		
Upper	if (EPRC $_\alpha$ ) infeasible then $\bar{u} \leftarrow \alpha$	if (SPRC $_\alpha$ ) infeasible then $\bar{u} \leftarrow \alpha$	
Lower	$S_{\text{MIP}} \leftarrow \{j \in J \mid \delta^*((x, -1)^T \mid \bar{U}_j) = 0\}, l \leftarrow \min_{j \in J \setminus S_{\text{MIP}}} \frac{\bar{b}^j - (\bar{a}^j)^T x}{\delta^*((x, -1)^T \mid \bar{U}_j)}$		
Results	$l$	$(l, \text{non optimal})$	

*Proof.* If we replace the operation Lower of **ScalingBin**, see Table 1, by  $l = \alpha$ , then **ScalingBin** equals a classic binary search. Thus, we have to prove that the outcome  $\alpha'$  of Lower satisfies  $\alpha \leq \alpha' \leq \bar{u}$  and that Problem (EPRC $_{\alpha'}$ ) is feasible. From the proof of Theorem 3.7 it follows the inequality  $\alpha \leq \alpha'$  and the feasibility of Problem (EPRC $_{\alpha'}$ ). Consequently,  $\alpha' \leq \bar{u}$  holds due to the monotonicity of (EPRC $_\alpha$ ) w.r.t.  $\alpha$  and  $\bar{u}$  being an upper bound for  $\rho_{\text{MIP}}$ .  $\square$

We note that if the RRF is attained by the solution  $x$  in the operation Subp, then **ScalingBin** directly scales the lower bound  $l$  to the RRF in the operation Lower, which is shown in the following lemma.

**Lemma 4.14.** *Let the RRF  $\rho_{\text{MIP}}$  of (P) be attained and  $0 \leq \alpha \leq \rho_{\text{MIP}}$ . Additionally, let  $x$  be a feasible solution to (EPRC $_\alpha$ ) as well as to (EPRC $_{\rho_{\text{MIP}}}$ ) and  $S_{\text{MIP}} = \{j \in J \mid \delta^*((x, -1)^T \mid \bar{U}_j) = 0\}$ . Then,  $\min_{j \in J \setminus S_{\text{MIP}}} \frac{\bar{b}^j - (\bar{a}^j)^T x}{\delta^*((x, -1)^T \mid \bar{U}_j)} = \rho_{\text{MIP}}$  holds.*

*Proof.* The claim follows in analogy to the proof of Theorem 3.7.  $\square$

We now integrate in **ScalingBin** the optimality condition for the RRF of Lemma 3.8 and Theorem 3.9 as an additional termination condition. Algorithm **MaxScalingBin**

preserves the properties of `ClassicBin` for every feasible solution  $x$  of  $(\mathbf{P})$  under the additional assumption  $\delta^*((x, -1)^T \mid \bar{U}_j) > 0$  for  $j \in J$ . Furthermore, in the case that the RRF is attained `MaxScalingBin` immediately stops if the RRF is computed. The latter is not guaranteed in `ScalingBin` because it possibly has to tighten the upper bound first before it stops. In order to avoid this effect, `MaxScalingBin` solves Problem  $(\text{SPRC}_\alpha)$  to optimality in every iteration whereas `ScalingBin` only checks the feasibility of  $(\text{EPRC}_\alpha)$  in every iteration. We note that the computational complexities of  $(\text{SPRC}_\alpha)$  and  $(\text{EPRC}_\alpha)$  are equal.

**Lemma 4.15.** *Let the inequalities  $\delta^*((x, -1)^T \mid \bar{U}_j) > 0$  for  $j \in J \setminus S_{\text{MIP}}$  hold for every feasible solution  $x$  of  $(\mathbf{P})$ . Let  $\rho_{\text{MIP}} \in \mathbb{R}$  be the finite RRF of  $(\mathbf{P})$  and  $(\alpha, \text{flag})$  the output of `MaxScalingBin`. Then,  $(\text{EPRC}_\alpha)$  is feasible. Additionally, if  $\text{flag}$  is equal to `optimal`, then  $\alpha = \rho_{\text{MIP}}$ , otherwise,  $|\rho_{\text{MIP}} - \alpha| \leq \text{tol}$  holds. Furthermore, `MaxScalingBin` performs at most  $\lceil \log_2(\frac{\bar{u}}{\text{tol}}) \rceil$  many iterations.*

*Proof.* Problem  $(\text{SPRC}_\alpha)$  is feasible if and only if  $(\text{EPRC}_\alpha)$  is feasible. Consequently, if `MaxScalingBin` returns  $(\alpha, \text{non optimal})$  the claim follows from Lemma 4.13. Otherwise, the claim follows from Lemma 3.8.  $\square$

Finally, we present an approach that is similar to `MaxScalingBin` and needs the same assumption, i.e., the inequalities  $\delta^*((x, -1)^T \mid \bar{U}_j) > 0$  for  $j \in J$  hold for every feasible solution  $x$  of  $(\mathbf{P})$ . The method, given by Algorithm `PureScaling`, is based on computing the maximal slack in each iteration and then scaling the current value of the RRF. The main goal is that if we get close to the RRF very fast, then we can detect this without tightening the upper bound of the RRF in many iterations such as it can happen in the previous presented approaches. We note that `PureScaling` is not based on a binary search. Additionally, the upper bound of the RRF is only necessary to guarantee a finite runtime.

**Lemma 4.16.** *Let the inequalities  $\delta^*((x, -1)^T \mid \bar{U}_j) > 0$  for  $j \in J \setminus S_{\text{MIP}}$  hold for every feasible solution  $x$  of  $(\mathbf{P})$ . Let  $(\alpha, \text{flag})$  be the output of `PureScaling` and  $\rho_{\text{MIP}} \in \mathbb{R}$  the finite RRF of  $(\mathbf{P})$ . Then,  $(\text{EPRC}_\alpha)$  is feasible. Additionally, if  $\text{flag}$  is equal to `optimal`, then  $\alpha = \rho_{\text{MIP}}$ , otherwise,  $|\rho_{\text{MIP}} - \alpha| \leq \text{tol}$  holds. Furthermore, `PureScaling` performs at most  $\lceil \frac{\rho_{\text{MIP}}}{\text{tol}} \rceil$  many iterations.*

*Proof.* The claim follows from Lemma 4.15 and the construction of `PureScaling`.  $\square$

We note that the worst-case runtime of `PureScaling` is inferior to the worst-case runtime of the presented approaches based on binary search. But in practice `PureScaling` detects faster if the computed RRF is in the a priori given tolerance than the approaches based on binary search, which we investigate experimentally in the next section.

## 5. COMPUTATIONAL RESULTS

In this section, we present a computational study for the previously described methods to compute the RRF for MIPs of the MIPLIB 2017 library, see [30]. To be more precise, we evaluate the impact of the aspects:

- (a) The chosen method: We compare the bounded IP approach (26), the classic binary search, and Methods `ScalingBin`, `MaxScalingBin`, and `PureScaling`.
- (b) The performance: We compare the runtime of every method and the corresponding number of iterations.
- (c) Characterization of the instances: We analyze the instances w.r.t. their computed RRF  $\rho_{\text{MIP}}$  and the impact of the uncertainties. In particular, we compare the optimal nominal objective value to the optimal objective value of  $(\text{PR}_{\rho_{\text{MIP}}})$ . This comparison quantifies the *price of robustness*.

We implemented the algorithms in Python 3.6.5 and solved the MIPs with Gurobi 8.0.1, see [26]. All computations were executed on a 4-core machine with a Xeon E3-1240 v5 CPU and 32 GB RAM. Our test set consists of 165 instances from the MIPLIB 2017 library. Out of the entire MIPLIB 2017 library of 1065 instances, we only considered the benchmark set of 240 instances. We further excluded 38 instances that are classified as hard in the MIPLIB 2017. This guarantees that we can solve the nominal problem by state-of-the-art available programs within a reasonable runtime. Additionally, we excluded the remaining 5 infeasible instances. We next determine the types of constraints that we consider as syntactically safe, i.e., these constraints have an uncertainty set consisting only of the zero vector. First, we consider every constraint that consists just of a single variable as safe because it directly represents a lower bound of the corresponding variable. Additionally, every constraint that contains only binary variables with coefficients  $\pm 1$  is safe because these constraints usually represent combinatorial structures. Considering the latter constraints as unsafe leads to infeasibility in most of the cases, i.e., the RRF is zero. Due to the same reason, we consider equalities as safe. In doing so, we also exclude equalities that are simply rewritten as two linear inequalities. No further presolve routines for detecting implicit equalities are processed. Considering the previously mentioned constraints as safe leads to 32 instances that only contain safe constraints. Consequently, these instances are also excluded, which finally results in our test set of 165 instances.

In all computations, we used Gurobi with standard settings with the following adaptations. For all methods, we disabled dual reductions in order to have a more definitive conclusion about infeasibility of the model. For the classical binary search and [ScalingBin](#), we set the parameter solution limit to 1 because we are only interested in the feasibility of the corresponding MIP in every iteration. In contrast to this, we solve the upcoming MIPs in every iteration of [MaxScalingBin](#) and [PureScaling](#) to optimality. In order to prevent that the extended runtime of solving these MIPs to optimality exceeds the potential benefit of maximizing the slack together with scaling the RRF, described in [MaxScalingBin](#) and [PureScaling](#), we set the relative MIP gap to 0.5 as this value turned out to be reasonable in our preliminary computational results. We consider an absolute tolerance of  $10^{-4}$  and set the time limit to 2 h. Furthermore, we introduced a relative tolerance of  $10^{-4}$  as an additional termination condition in order to avoid numerical issues.

We next turn to the considered uncertainty set. We compute the RRF in the extended framework of Section 3. Consequently, the  $j$ th unsafe constraint has the uncertainty set  $\bar{U}_j := \mu^j U_j \subset \mathbb{R}^{n+1}$ , composed of a convex and compact set  $U_j$  that is scaled by the smallest absolute nonzero coefficient  $\mu^j$  of the  $j$ th constraint. In our computational study, the uncertainty set  $U_j$  for the  $j$ th unsafe constraint is given as follows. For each variable with nonzero coefficient in the  $j$ th constraint, the uncertainty set for this variable is given by the interval  $[-1, 1]$ . The latter interval is also the uncertainty set of the right-hand side. If a variable has coefficient zero in the considered constraint, then it is considered safe for this constraint, i.e., its corresponding uncertainty set contains only zero. In total, the uncertainty set  $U_j$  is given by the corresponding cross products of intervals  $[-1, 1]$  and sets  $\{0\}$ . For  $\bar{U}_j \neq \{0\}$ , it follows from the construction of  $\bar{U}_j$  that the  $(n+1)$ th unit vector of  $\mathbb{R}^{n+1}$  is in  $\bar{U}_j$  and thus,  $\delta^*((x, -1) \mid \bar{U}_j) > 0$  holds for every  $x \in \mathbb{R}^n$ . Consequently, each constraint with uncertainty set unequal to zero is (semantically) unsafe.

We next turn to the computation of an upper bound of the RRF w.r.t. the considered uncertainty set, which is necessary for the proposed methods. For the

chosen uncertainty set, an upper bound  $\bar{u}$  of the RRF for (P) is given by

$$\bar{u} = \min_{j \in K} \frac{\max\{0, \bar{b}_j, \max\{|\bar{a}_i^j| : i = 1, \dots, n\}\}}{\mu^j},$$

whereby  $K$  is the index set of the unsafe constraints. The value  $\bar{u}$  is an upper bound for the RRF due to the following short explanation. If we assume that the RRF  $\alpha$  satisfies  $\alpha > \bar{u}$ , then an index  $k \in K$  exists such that  $\pm \bar{a}_i^k \in \alpha \bar{U}_{k_i}$  for  $i \in \{1, \dots, n\}$  and  $-(\max\{0, \bar{b}^k\} + \varepsilon) \in \alpha \bar{U}_{k_{\bar{b}}}$  for a sufficient small  $\varepsilon > 0$  holds. Consequently, for every solution  $x$  a realization  $u \in \bar{U}_k$  exists such that

$$(\bar{a}^k)^T x + \alpha((u_I)^T x - u_{\bar{b}}) - \bar{b}^k > 0$$

holds, which directly implies the infeasibility of (EPR $_{\alpha}$ ).

We now turn to the presentation and discussion of the numerical results. We note that we excluded the bounded IP approach in this numerical analysis because preliminary results showed that its performance is massively worse when compared to the other proposed methods. The performance of the proposed methods might differ between instances with positive RRF and instances with RRF zero. For example, if the RRF is zero, then only [PureScaling](#) automatically terminates after a single iteration independent from the chosen MIP gap. Consequently, we will separately analyze the numerical results for instances with positive RRF and with RRF zero. According to our results, the considered 165 instances split into the following sets: 66 instances with positive RRF, 85 instances with RRF zero, 13 instances which could not be solved in the timelimit of 2 h by any method, and one instance (`rmatr100-p10`) which could not be solved due to numerical issues. We now use log-scaled performance profiles to compare runtimes as proposed in [21]. We note that all runtimes include the computation of the upper bound. Figure 1 shows the performance profiles for instances with positive RRF and Figure 2 for instances with RRF zero. Furthermore, a short statistical summary of the runtimes and number of iterations is given in Tables 2 and 3. Overall, we see that the performance of the classical binary search and [ScalingBin](#) is nearly the same, independent from the RRF values. For instances with positive RRF, we see that the classical binary search, [ScalingBin](#), and [PureScaling](#) solve the same number of instances, 97% overall, while [MaxScalingBin](#) solves one instance less. In doing so, the best performance is given by [PureScaling](#) which outperforms the remaining methods. The performance of [MaxScalingBin](#) follows which is slightly better in comparison to the classic binary search, respectively [ScalingBin](#). For instances with RRF zero, we recognize a similar performance pattern. This time the performances of [MaxScalingBin](#) and [PureScaling](#) are nearly identical and they outperform the other approaches in most of the cases. This improved performance of [MaxScalingBin](#) and [PureScaling](#) for instances with RRF zero is mainly explained by the fact that both algorithms almost always terminate after the first iteration, see Table 3. We note that this behavior is not necessarily guaranteed for [MaxScalingBin](#) in contrast to [PureScaling](#) due to the chosen MIP gap. However, the numerical results show that in most of the cases (SPRC $_{\alpha}$ ) is solved to optimality in the first iteration, i.e., the corresponding objective value is zero. Consequently, a RRF of zero is immediately detected. [MaxScalingBin](#) as well as [PureScaling](#) solve 96% of the instances with RRF zero, whereas the classic binary search and [ScalingBin](#) solve all of these instances. The effect that the latter two approaches solve slightly more instances can be explained by the fact that both methods only check feasibility in every iteration instead of solving the corresponding MIPs to optimality as in [MaxScalingBin](#) and [PureScaling](#). Generally, the latter is more time-consuming.

Considering the number of iterations, maximizing the slack together with scaling the RRF, as proposed in [MaxScalingBin](#) and [PureScaling](#), significantly reduces the

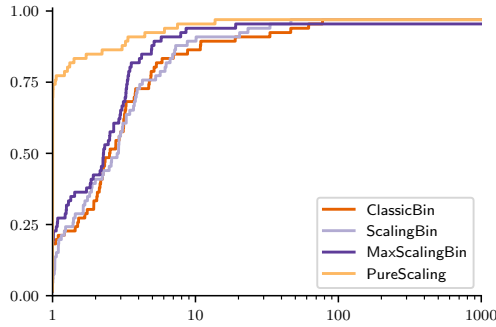


FIGURE 1. Log-scaled performance profiles of runtimes for instances with positive RRF

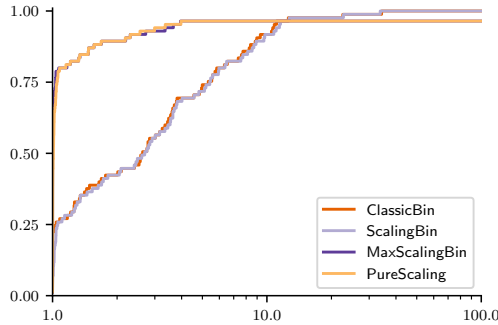


FIGURE 2. Log-scaled performance profiles of runtimes for instances with RRF zero

number of necessary iterations. In general, the number of iterations in **MaxScalingBin** is higher than in **PureScaling** because the latter checks in every iteration if the computed RRF is already in tolerance and does not have to lower an upper bound such as **MaxScalingBin**. Furthermore, it is interesting to see that scaling the RRF without maximizing the slack, as we do in **ScalingBin**, does not significantly decrease the necessary number of iterations, respectively the runtime, in comparison with the classical binary search. Furthermore, the statistical parameters of Tables 2 and 3 indicate that in most cases the RRF can be computed quickly ( $< 60$  s). Only for a minority of the instances the runtimes drastically increase.

Based on the previous analysis of the results, we suggest to compute the RRF of a MIP as follows. First, run **PureScaling** with a small time or iteration limit. If the RRF could not be computed within the set limit, then we suggest to switch to the classical binary search, respectively to **ScalingBin**, because these methods solve more instances overall.

Finally, we turn to a short discussion about the price of robustness. To this end, we compare the optimal objective value of the nominal problem (P) and of the robust problem ( $\text{PR}_{\rho_{\text{MIP}}}$ ). In the time limit of 2 h, we could optimally solve 51 of the 66 Problems ( $\text{PR}_{\rho_{\text{MIP}}}$ ) with positive RRF. We then computed the price of robustness  $p$  as follows. Let the value  $w$  be the optimal nonzero objective value of (P) and  $w^*$  of ( $\text{PR}_{\rho_{\text{MIP}}}$ ). Then, the price of robustness is given by  $p = \frac{w^* - w}{|w|}$ . As we can see in Table 4, the price of robustness is subject to strong fluctuations. On

TABLE 2. Number of solved instances (out of 66 instances with positive RRF) and statistics for the runtimes and number of iterations (always taken only for all instances solved to optimality)

	ClassicBin		ScalingBin		MaxScalingBin		PureScaling	
#solved	64		64		63		64	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
Minimum	0.23	15.00	0.13	1.00	0.11	1.00	0.11	1.00
1st Quartile	1.56	15.00	1.69	15.00	1.48	10.50	0.63	1.00
Median	7.30	15.00	9.20	15.00	6.85	14.00	3.35	1.00
Mean	315.12	17.50	259.72	16.13	196.18	13.38	251.98	3.22
3rd Quartile	82.98	18.00	83.27	17.00	42.59	17.00	42.67	3.00
Maximum	5800.64	32.00	6857.40	32.00	4276.22	32.00	4622.70	29.00

TABLE 3. Number of solved instances (out of 85 instances with RRF zero) and statistics for the runtimes and number of iterations (always taken only for all instances solved to optimality)

	ClassicBin		ScalingBin		MaxScalingBin		PureScaling	
#solved	85		85		82		82	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
Minimum	0.26	15.00	0.32	15.00	0.17	1.00	0.16	1.00
1st Quartile	5.69	15.00	5.95	15.00	2.14	1.00	2.15	1.00
Median	23.46	15.00	24.03	15.00	6.69	1.00	6.84	1.00
Mean	199.03	16.41	201.30	16.41	117.00	1.35	115.28	1.00
3rd Quartile	102.61	16.00	105.57	16.00	29.74	1.00	29.60	1.00
Maximum	4447.14	35.00	4490.15	35.00	2565.66	16.00	2567.18	1.00

the one hand instances with a small or even zero price of robustness exist. On the other hand for some instances the robustness of the solution comes along with an immense deterioration of the objective value. Surprisingly, the median shows that for many instances the price of robustness is in a reasonable limit keeping in mind that the considered uncertainty set has its maximal size w.r.t. robust feasibility. The results illustrate that choosing the “most robust” solution, as proposed in Section 1, does not necessarily come along with a high price of robustness. Furthermore, the price of robustness can be limited a priori by a so called budget constraint that is often desired in applications, see Section 1. Overall, the RRF can be useful as a decision rule to decide between different robust optimal solutions w.r.t. the size of the uncertainty set. We further note that the numerical results do not indicate relations between the percentage of unsafe constraints, the size of the RRF, and the price of robustness.

In practice, a decision maker often faces the following bi-objective challenge: On the one hand, one aims at guaranteeing robust feasibility of an optimal solution for the largest possible uncertainty set  $\alpha \bar{U}_j, j \in J$ , i.e., one wants to maximize  $\alpha \in [0, \rho_{\text{MIP}}]$ , respectively  $\alpha \in [0, \rho_{\text{MIP}}[$  if the RRF is not attained. On the other hand, however, one wants to minimize the optimal value of the robust counterpart or, equivalently, the price of robustness, which usually comes with a smaller  $\alpha$ . Consequently, a trade-off between robustness and minimum cost has to be made. We



TABLE 4. Statistics for the best computed RRF and price of robustness (always taken only for all instances solved to optimality).

	Unsafe Constraints (%)	RRF	Price of Robustness (%)
Minimum	0.003	0.0001	0.00
1st Quartile	4.118	0.6200	93.28
Median	37.848	0.9901	384.16
Mean	43.015	67.0442	29 368 526 813.18
3rd Quartile	83.777	1.1509	20 877.96
Maximum	100.000	1006.0000	1 297 693 684 636.35

exemplarily illustrate three different characteristics for this trade-off in Figures 3-5, that we found in our computational experiments. To this end, we first discretized the interval  $[0, \rho_{\text{MIP}}]$  equidistantly and then computed the optimal value of the robust counterpart for each of these points. From Figure 3, it can be concluded that an increase in robustness comes with increasing cost, i.e., the price of robustness increases. Here, the trade-off between robustness and the optimal value is quite regular, i.e., for a possibly small increase of robustness, we always find a solution with a modest increase of cost. In contrast to this, we have a stepwise effect in Figure 4. Here, increasing the robustness can lead to two different effects regarding the cost. On the one hand, an increase of robustness can have almost no effect on the cost, which is for example the case for the interval  $(0, 0.20)$ . But on the other hand, pushing the robustness above a certain level, even by a really small increase, can lead to a very large increase of the cost, which is for example the case for a robustness level of at least 0.20. In Figure 5, both previously mentioned effects between robustness and cost exist. For  $\alpha \in [0, 0.35]$ , an increase of the robustness comes with larger cost, i.e., the price of robustness increases. In contrast to this, for  $\alpha \in [0.35, 0.8509]$  an increase of the robustness comes with no or a modest increase of the optimal value of the robust counterpart.

Overall, an increase of robustness usually comes with an increase of the optimal value, i.e. the price of robustness increases. But as Figure 4 and 5 show, sometimes there is a possibility to significantly increase the robustness for small or no cost.

## 6. CONCLUSION

In this paper, we studied the problem of finding the “maximal” size of a given uncertainty set for a MIP such that its robust feasibility is guaranteed. In doing so, we determined this maximal size with the help of the radius of robust feasibility (RRF). We first motivated the investigations of this paper. We introduced the RRF for MIPs and then analyzed it w.r.t. its LP relaxation in the common setting of the literature. The latter requires a full-dimensional uncertainty set and thus, every variable is “unsafe”. In particular, we proved that the RRF of a MIP and of its LP relaxation equal if the RRF of the relaxation is not attained. In special cases, this allows us to compute the RRF of a MIP with known techniques for the RRF of LPs. In order to make the RRF applicable to a broader spectrum of optimization problems, we extended the common setting of the RRF such that the uncertainty set is not necessarily full-dimensional and potentially different for every constraint. This allows to model safe variables and constraints, which are not affected by any uncertainty. We then proposed methods for computing the RRF of linear as well as mixed-integer problems in our extended setting. These methods can be seen as a first benchmark for computing the RRF including safe variables and constraints.

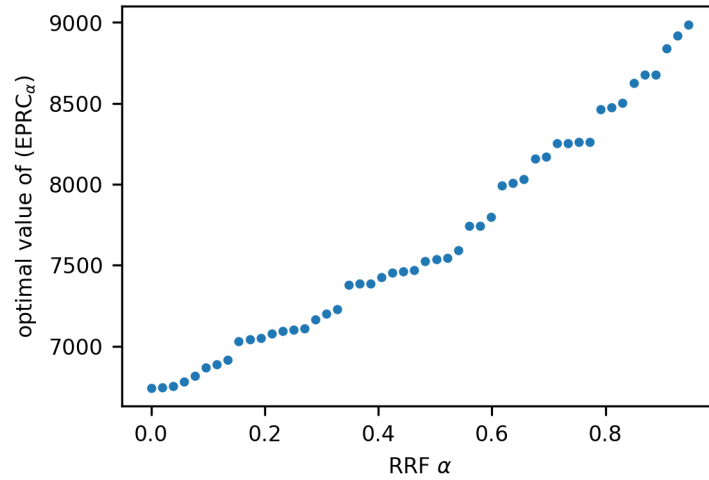


FIGURE 3. Trade-off between robustness and cost, i.e., optimal value of robust counterpart ( $\text{EPRC}_{\alpha}$ ) as a function of the size of  $\alpha \bar{U}_j, j \in J$ , for instance `binkar10_1` with RRF  $\rho_{\text{MIP}} = 0.9459$ .

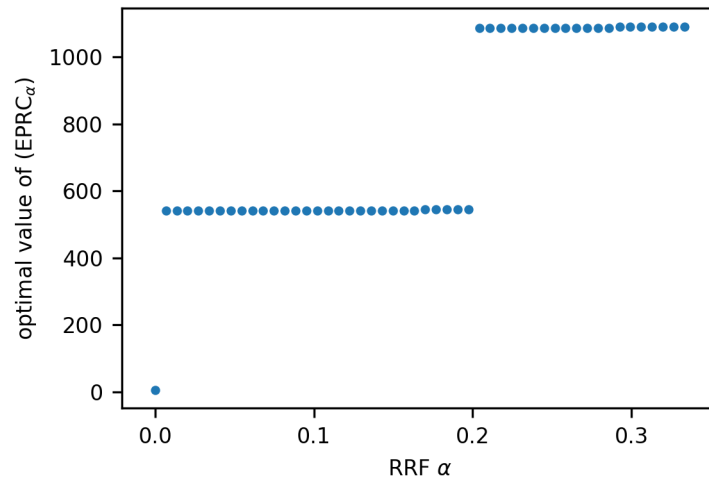


FIGURE 4. Trade-off between robustness and cost, i.e., optimal value of robust counterpart ( $\text{EPRC}_{\alpha}$ ) as a function of the size of  $\alpha \bar{U}_j, j \in J$ , for instance `comp07-2idx` with RRF  $\rho_{\text{MIP}} = 0.3333$ .

Finally, we illustrated the applicability of our methods by computing the RRF for MIPs of the MIPLIB 2017 library.

Further research and methods for computing the RRF in the extended framework are desirable, especially for a comparison with our methods. Also the extended RRF can now be applied to compute the “most robust” solution within an a priori budget for different applications. Additionally, it seems promising to use the information about the “maximal” size of an uncertainty set, computed by the RRF, in order to construct suitable uncertainty sets for robust optimization models. Moreover, sizing uncertainty sets w.r.t. alternative concepts of robustness, e.g., adjustable robustness, plays an important role in many applications: e.g., in gas networks

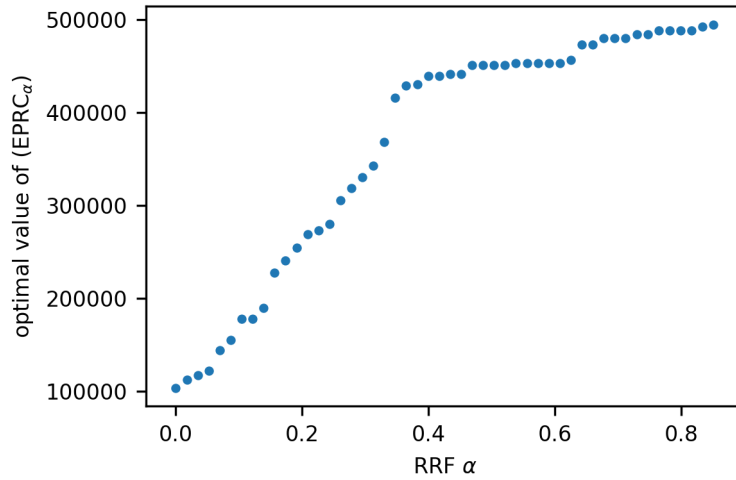


FIGURE 5. Trade-off between robustness and cost, i.e., optimal value of robust counterpart ( $\text{EPRC}_\alpha$ ) as a function of size of  $\alpha \bar{U}_j, j \in J$ , for instance drayage-100-23 with  $\text{RRF } \rho_{\text{MIP}} = 0.8509$ .

it can be used for validating the feasibility of a booking [32] and for the optimal operation under technical uncertainties [1]. Thus, introducing the RRF for other concepts of robustness, especially adjustable robustness, are interesting topics for future research.

#### ACKNOWLEDGMENTS

This research has been performed as part of the Energie Campus Nürnberg and is supported by funding of the Bavarian State Government. The first and second author also thank the DFG for their support within projects B06 and B07 in the CRC TRR 154. We want to thank Ruth Misener for pointing out the relation between the RRF and the flexibility index problem. Finally, we want to thank an anonymous reviewer for thoroughly reading a former version of the paper and his/her detailed comments that helped to improve the paper.

#### REFERENCES

- [1] D. Aßmann, F. Liers, and M. Stingl. “Decomposable robust two-stage optimization: An application to gas network operations under uncertainty.” In: *Networks* (Jan. 2019). DOI: [10.1002/net.21871](https://doi.org/10.1002/net.21871).
- [2] M. Avriel, W. Diewert, S. Schaible, and I. Zang. *Generalized Concavity*. SIAM, 2010. DOI: [10.1137/1.9780898719437](https://doi.org/10.1137/1.9780898719437).
- [3] A. Ben-Tal, L. El Ghaoui, and A. Nemirovski. *Robust Optimization*. Princeton University Press, Princeton, NJ, 2009. DOI: [10.1515/9781400831050](https://doi.org/10.1515/9781400831050).
- [4] A. Ben-Tal, D. den Hertog, and J.-P. Vial. “Deriving robust counterparts of nonlinear uncertain inequalities.” In: *Math. Program.* 149.1-2, Ser. A (2015), pp. 265–299. DOI: [10.1007/s10107-014-0750-8](https://doi.org/10.1007/s10107-014-0750-8).
- [5] A. Ben-Tal and A. Nemirovski. “Selected topics in robust convex optimization.” In: *Math. Program.* 112.1, Ser. B (2008), pp. 125–158. DOI: [10.1007/s10107-006-0092-2](https://doi.org/10.1007/s10107-006-0092-2).

- [6] D. Bertsimas and D. B. Brown. “Constructing uncertainty sets for robust linear optimization.” In: *Oper. Res.* 57.6 (2009), pp. 1483–1495. DOI: [10.1287/opre.1080.0646](https://doi.org/10.1287/opre.1080.0646).
- [7] D. Bertsimas, D. B. Brown, and C. Caramanis. “Theory and applications of robust optimization.” In: *SIAM Rev.* 53.3 (2011), pp. 464–501. DOI: [10.1137/080734510](https://doi.org/10.1137/080734510).
- [8] D. Bertsimas, V. Gupta, and N. Kallus. “Data-driven robust optimization.” In: *Math. Program.* 167.2, Ser. A (2018), pp. 235–292. DOI: [10.1007/s10107-017-1125-8](https://doi.org/10.1007/s10107-017-1125-8).
- [9] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, 2004. DOI: [10.1017/CB09780511804441](https://doi.org/10.1017/CB09780511804441).
- [10] C. Buchheim and J. Kurtz. “Robust combinatorial optimization under convex and discrete cost uncertainty.” In: *EURO J. Comput. Optim.* 6.3 (2018), pp. 211–238. DOI: [10.1007/s13675-018-0103-0](https://doi.org/10.1007/s13675-018-0103-0).
- [11] M. J. Cánovas, M. A. López, J. Parra, and M. I. Todorov. “Stability and well-posedness in linear semi-infinite programming.” In: *SIAM J. Optim.* 10.1 (1999), pp. 82–98. DOI: [10.1137/S1052623497319869](https://doi.org/10.1137/S1052623497319869).
- [12] M. J. Cánovas, M. A. López, J. Parra, and F. J. Toledo. “Distance to ill-posedness and the consistency value of linear semi-infinite inequality systems.” In: *Mathematical Programming* 103.1 (2005), pp. 95–126. DOI: [10.1007/s10107-004-0519-6](https://doi.org/10.1007/s10107-004-0519-6).
- [13] M. J. Cánovas, M. A. López, J. Parra, and F. J. Toledo. “Distance to ill-posedness for linear inequality systems under block perturbations: convex and infinite-dimensional cases.” In: *Optimization* 60.7 (2011), pp. 925–946. DOI: [10.1080/02331934.2011.606624](https://doi.org/10.1080/02331934.2011.606624).
- [14] E. Carrizosa and S. Nickel. “Robust facility location.” In: *Mathematical Methods of Operations Research* 58.2 (Nov. 2003), pp. 331–349. DOI: [10.1007/s001860300294](https://doi.org/10.1007/s001860300294).
- [15] E. Carrizosa, A. Ushakov, and I. Vasilyev. “Threshold robustness in discrete facility location problems: a bi-objective approach.” In: *Optim. Lett.* 9.7 (2015), pp. 1297–1314. DOI: [10.1007/s11590-015-0892-5](https://doi.org/10.1007/s11590-015-0892-5).
- [16] T. C. Chan and P. A. Mar. “Stability and continuity in robust optimization.” In: *SIAM Journal on Optimization* 27.2 (2017), pp. 817–841. DOI: [10.1137/16M1067512](https://doi.org/10.1137/16M1067512).
- [17] A. Charnes and W. W. Cooper. “Programming with linear fractional functionals.” In: *Naval Res. Logist. Quart.* 9 (1962), pp. 181–186. DOI: [10.1002/nav.3800090303](https://doi.org/10.1002/nav.3800090303).
- [18] J. Chen, J. Li, X. Li, Y. Lv, and J.-C. Yao. “Radius of robust feasibility of system of convex inequalities with uncertain data.” In: *J. Optim. Theory Appl.* 184.2 (2020), pp. 384–399. DOI: [10.1007/s10957-019-01607-7](https://doi.org/10.1007/s10957-019-01607-7).
- [19] T. D. Chuong and V. Jeyakumar. “An exact formula for radius of robust feasibility of uncertain linear programs.” In: *J. Optim. Theory Appl.* 173.1 (2017), pp. 203–226. DOI: [10.1007/s10957-017-1067-6](https://doi.org/10.1007/s10957-017-1067-6).
- [20] G. P. Crespi, D. Kuroiwa, and M. Rocca. “Robust optimization: Sensitivity to uncertainty in scalar and vector cases, with applications.” In: *Operations Research Perspectives* 5 (2018), pp. 113–119. DOI: [10.1016/j.orp.2018.03.001](https://doi.org/10.1016/j.orp.2018.03.001).
- [21] E. D. Dolan and J. J. Moré. “Benchmarking optimization software with performance profiles.” In: *Math. Program.* 91.2, Ser. A (2002), pp. 201–213. DOI: [10.1007/s101070100263](https://doi.org/10.1007/s101070100263).

- [22] M. A. Goberna, V. Jeyakumar, G. Li, and N. Linh. “Radius of robust feasibility formulas for classes of convex programs with uncertain polynomial constraints.” In: *Oper. Res. Lett.* 44.1 (2016), pp. 67–73. DOI: [10.1016/j.orl.2015.11.011](https://doi.org/10.1016/j.orl.2015.11.011).
- [23] M. A. Goberna, V. Jeyakumar, G. Li, and J. Vicente-Pérez. “Robust solutions of multiobjective linear semi-infinite programs under constraint data uncertainty.” In: *SIAM J. Optim.* 24.3 (2014), pp. 1402–1419. DOI: [10.1137/130939596](https://doi.org/10.1137/130939596).
- [24] M. A. Goberna, V. Jeyakumar, G. Li, and J. Vicente-Pérez. “Robust solutions to multi-objective linear programs with uncertain data.” In: *European J. Oper. Res.* 242.3 (2015), pp. 730–743. DOI: [10.1016/j.ejor.2014.10.027](https://doi.org/10.1016/j.ejor.2014.10.027).
- [25] B. L. Gorissen, İ. Yanıkoğlu, and D. den Hertog. “A practical guide to robust optimization.” In: *Omega* 53 (2015), pp. 124–137. DOI: [10.1016/j.omega.2014.12.006](https://doi.org/10.1016/j.omega.2014.12.006).
- [26] Gurobi Optimization, LLC. *Gurobi optimizer reference manual*. 2018. URL: <http://www.gurobi.com>.
- [27] J.-B. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms. I*. Vol. 305. Grundlehren der Mathematischen Wissenschaften. Fundamentals. Springer-Verlag, Berlin, 1993, pp. xviii+417. DOI: [10.1007/978-3-662-02796-7](https://doi.org/10.1007/978-3-662-02796-7).
- [28] T. Koch, B. Hiller, M. Pfetsch, and L. Schewe, eds. *Evaluating Gas Network Capacities*. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2015. DOI: [10.1137/1.9781611973693](https://doi.org/10.1137/1.9781611973693).
- [29] X.-B. Li and Q.-L. Wang. “A note on the radius of robust feasibility for uncertain convex programs.” In: *Filomat* 32 (Jan. 2018), pp. 6809–6818. DOI: [10.2298/FIL1819809L](https://doi.org/10.2298/FIL1819809L).
- [30] *MIPLIB 2017*. <http://miplib.zib.de>. 2018.
- [31] S. Schaible. “Fractional programming, I. Duality.” In: *Management Sci.* 22.8 (1976), pp. 858–867. DOI: [10.1287/mnsc.22.8.858](https://doi.org/10.1287/mnsc.22.8.858).
- [32] L. Schewe, M. Schmidt, and J. Thürauf. “Structural properties of feasible bookings in the european entry-exit gas market system.” In: *4OR* 18.2 (2020), pp. 197–218. DOI: [10.1007/s10288-019-00411-3](https://doi.org/10.1007/s10288-019-00411-3).
- [33] Q. Zhang, I. E. Grossmann, and R. M. Lima. “On the relation between flexibility analysis and robust optimization for linear systems.” In: *AIChE Journal* 62.9 (2016), pp. 3109–3123. DOI: [10.1002/aic.15221](https://doi.org/10.1002/aic.15221).

(F. Liers, J. Thürauf) <sup>1</sup>FRIEDRICH-ALEXANDER-UNIVERSITÄT ERLANGEN-NÜRNBERG, DISCRETE OPTIMIZATION, CAUERSTR. 11, 91058 ERLANGEN, GERMANY; <sup>2</sup>ENERGIE CAMPUS NÜRNBERG, FÜRTHSTR. 250, 90429 NÜRNBERG, GERMANY

*E-mail address:* {[frauke.liers](mailto:frauke.liers@fau.de), [johannes.thuerauf](mailto:johannes.thuerauf@fau.de)}@fau.de

(L. Schewe) <sup>3</sup>THE UNIVERSITY OF EDINBURGH, SCHOOL OF MATHEMATICS, JAMES CLERK MAXWELL BUILDING, PETER GUTHRIE TAIT ROAD, EDINBURGH, EH9 3FD, UK

*E-mail address:* {[lars.schewe](mailto:lars.schewe@ed.ac.uk)}@ed.ac.uk

## APPENDIX. DETAILED NUMERICAL RESULTS

In this section, we provide the detailed numerical results of our computational study. We list all instances for which we could not compute the RRF in our time limit of 2 h in Table 5. For instances with positive RRF, the best computed RRF, the percentage of unsafe constraints, and its price of robustness are given in Table 6. The detailed runtimes and number of iterations for instances with positive RRF are given in Table 7 and for instances with RRF zero in Table 8.

TABLE 5. Summary of instances that hit the timelimit of 2 h or could not be solved due to numerical issues (rmatr100-p10)

Instances	Unsafe Constraints (%)	RRF
cryptanalysisiskb128n5obj16	44.950	–
glass4	90.909	–
neos-1456979	96.750	–
neos-2746589-doon	48.436	–
neos-3004026-krka	66.321	–
neos-3024952-loue	97.166	–
neos-3046615-murg	51.807	–
neos-3381206-awhea	99.165	–
neos-5107597-kakapo	98.938	–
nursesched-sprint02	0.568	–
rmatr100-p10	99.986	–
rmatr200-p5	99.997	–
supportcase26	95.402	–
swath3	42.986	–

TABLE 6. Summary price of robustness for instances with positive RRF (All values are standard rounded except of the RRF values. The latter values are rounded down in order to prevent an overestimation which can lead to an infeasibility of the corresponding robust counterpart. The computation of the price of robustness is based on the exact values of the numerical results)

Instances	Unsafe Constraints (%)	RRF	Nominal Objective	Robust Objective	Price of Robustness (%)
30n20b8	84.375	0.0156	302.00	604.00	100.00
50v-10	78.541	119.6930	3311.18	-	-
assign1-5-8	80.745	0.3846	212.00	-	-
binkar10_1	0.975	0.9459	6742.20	8986.79	33.29
bppc4-08	81.982	0.9999	53.00	-	-
brazil3	1.202	0.9999	24.00	-	-
buildingenergy	90.533	0.3975	33 283.85	782 078.32	2249.72
CMS750_4	66.821	0.0021	252.00	1000.00	296.83
comp07-2idx	0.513	0.3333	6.00	1091.00	18 083.30
cost266-UUE	3.942	0.5857	25 148 940.56	29 503 525.47	17.32
csched007	14.245	0.0296	351.00	590.40	68.20
csched008	14.245	0.9999	173.00	94 999 859.99	$5.49 \times 10^7$
drayage-100-23	91.253	0.8509	103 333.87	494 653.66	378.70
drayage-25-23	91.253	0.8509	101 282.65	490 365.10	384.16
fiball	12.895	0.2790	138.00	1729.00	1152.90
gen-ip002	100.000	241.2800	-4783.73	-41.79	99.13
gen-ip054	100.000	1.1641	6840.97	7 231 278 446.60	$1.06 \times 10^8$
gmu-35-40	3.774	0.1122	-2 406 733.37	-2 239 650.68	6.94
gmu-35-50	3.678	0.1122	-2 607 958.33	-	-
graphdraw-domain	39.306	0.1136	19 686.00	64 795.72	229.15
ic97_potential	91.013	0.0204	3942.00	4255.31	7.95
leo1	14.840	0.9333	404 227 536.16	-	-

Continued on next page

TABLE 6. Summary price of robustness for instances with positive RRF (All values are standard rounded except of the RRF values. The latter values are rounded down in order to prevent an overestimation which can lead to an infeasibility of the corresponding robust counterpart. The computation of the price of robustness is based on the exact values of the numerical results)

Instances	Unsafe Constraints (%)	RRF	Nominal Objective	Robust Objective	Price of Robustness (%)
leo2	24.789	0.9333	404077441.12	—	—
mad	21.569	16.3333	0.03	2.02	7437.31
mas74	92.308	0.9999	11801.19	118171839.07	$1.00 \times 10^6$
mas76	91.667	0.9999	40005.05	837260542319.45	$2.09 \times 10^9$
mc11	79.167	0.9814	11689.00	128960.00	1003.26
mik-250-20-75-4	38.462	951.0000	-52301.00	0.00	100.00
mushroom-best	99.953	0.9090	0.06	3916.90	$7.08 \times 10^6$
n3div36	1.293	2.9897	130800.00	—	—
n5-3	9.416	226.3543	8105.00	—	—
n9-3	7.107	218.5548	14409.00	—	—
neos-1122047	100.000	0.0001	161.00	162.00	0.62
neos-3083819-nubu	0.127	907.2763	6307996.00	7019780.00	11.28
neos-4763324-toguru	49.676	0.9999	1613.04	—	—
neos-662469	37.235	0.3333	184380.00	—	—
neos-787933	92.989	44.0000	30.00	1764.00	5780.00
neos-860300	60.000	6.6666	3201.00	3201.00	0.00
neos-911970	44.860	0.9999	54.76	232510087.68	$4.25 \times 10^8$
neos-933966	26.920	0.9999	318.00	4126665917461.59	$1.30 \times 10^{12}$
neos17	99.794	416.9255	0.15	1.62	979.91
neos5	95.238	0.7272	15.00	60.00	300.00
nexp-150-20-8-5	50.455	0.9749	231.00	—	—
ns1760995	0.032	0.7225	-549.21	-69.06	87.43

Continued on next page



TABLE 6. Summary price of robustness for instances with positive RRF (All values are standard rounded except of the RRF values. The latter values are rounded down in order to prevent an overestimation which can lead to an infeasibility of the corresponding robust counterpart. The computation of the price of robustness is based on the exact values of the numerical results)

Instances	Unsafe Constraints (%)	RRF	Nominal Objective	Robust Objective	Price of Robustness (%)
nu25-pr12	5.015	0.9999	53 905.00	212 500 011 900.00	$3.94 \times 10^8$
p200x1188c	85.591	1.9969	15 078.00	3 584 430.00	23 672.60
pg	20.000	1006.0000	-8674.34	7018.05	180.91
pk1	66.667	0.9999	11.00	7 641 184.01	$6.95 \times 10^7$
rai102	0.003	1.0000	-200.45	-199.86	0.29
ran14x18-disj-8	81.655	0.8666	3712.00	42 404.10	1042.35
reblock115	0.422	219.5121	-36 800 603.23	0.00	100.00
rococoB10-011000	5.399	0.8959	19 449.00	108 960.00	460.23
rococoC11-011100	4.647	0.9029	20 889.00	146 524.00	601.44
s250r10	0.146	1.1111	-0.17	-	-
satellites2-40	13.731	0.0001	-19.00	-16.00	15.79
satellites2-60-fs	17.389	0.0001	-19.00	-16.00	15.79
savsched1	0.886	0.9999	3217.70	3 858 535 013.84	$1.20 \times 10^8$
seymour1	86.064	0.3333	410.76	587.59	43.05
sp150x300d	66.667	5.1388	69.00	300.00	334.78
square41	0.090	0.9988	15.00	1681.00	11 106.70
square47	0.068	0.9990	16.00	2209.00	13 706.30
supportcase18	50.000	0.8666	48.00	120.00	150.00
supportcase42	100.000	0.9999	7.76	15 271 400 016.50	$1.97 \times 10^{11}$
swath1	42.986	0.7576	379.07	-	-
trento1	1.265	0.8552	5 189 487.00	9 993 978 424.72	192 481
var-smallemercy-m6j6	1.118	6.8910	-149.38	226.79	251.83

TABLE 7. Runtimes and number of iterations for instances with positive RRF (all values are standard rounded)

Instances	ClassicBin		ScalingBin		maxScalingBin		PureScaling	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
30n20b8	<b>13.62</b>	15	14.11	15	42.58	13	60.24	<b>6</b>
50v-10	1.56	16	1.70	16	0.70	10	<b>0.49</b>	<b>2</b>
assign1-5-8	80.56	17	80.81	17	13.69	17	<b>1.74</b>	<b>2</b>
binkar10_1	1.98	18	2.76	16	3.02	17	<b>0.92</b>	<b>3</b>
bppc4-08	3.92	20	4.87	20	3.86	20	<b>0.67</b>	<b>1</b>
brazil3	227.63	15	227.89	15	<b>6.85</b>	<b>1</b>	6.89	<b>1</b>
buildingenergy	<b>1903.99</b>	<b>15</b>	1975.59	<b>15</b>	> 7200	-	> 7200	-
CMS750_4	9.10	15	10.10	15	8.84	13	<b>3.93</b>	<b>2</b>
comp07-2idx	7.01	15	6.57	12	4.03	14	<b>4.01</b>	<b>2</b>
cost266-UUE	6.01	32	6.32	32	13.88	32	<b>1.62</b>	<b>1</b>
csched007	<b>346.43</b>	16	346.56	16	> 7200	-	2593.44	<b>4</b>
csched008	1.25	15	1.26	15	0.39	<b>1</b>	<b>0.38</b>	<b>1</b>
drayage-100-23	7.07	30	9.24	29	7.76	29	<b>3.16</b>	<b>3</b>
drayage-25-23	7.03	30	9.16	29	8.12	29	<b>3.24</b>	<b>3</b>
fiball	<b>23.28</b>	15	26.82	14	40.24	11	51.70	<b>10</b>
gen-ip002	0.25	15	0.13	2	<b>0.12</b>	<b>1</b>	0.13	<b>1</b>
gen-ip054	0.37	15	0.36	15	<b>0.22</b>	<b>11</b>	0.67	29
gmu-35-40	2449.44	15	13.61	15	13.51	14	<b>7.26</b>	<b>4</b>
gmu-35-50	3893.63	15	1464.00	15	<b>62.83</b>	11	211.56	<b>7</b>
graphdraw-domain	2052.99	15	2049.67	15	4276.22	11	<b>1695.64</b>	<b>5</b>
ic97_potential	90.23	21	90.63	21	43.12	18	<b>39.95</b>	<b>9</b>
leo1	26.07	15	35.58	14	15.57	12	<b>5.35</b>	<b>2</b>
leo2	97.17	15	121.69	14	41.17	12	<b>13.89</b>	<b>2</b>

Continued on next page

TABLE 7. Runtimes and number of iterations for instances with positive RRF (all values are standard rounded)

Instances	ClassicBin		ScalingBin		maxScalingBin		PureScaling	
	time/s	iter	time/s	iter	time/s	iter	time/s	iter
mad	0.26	15	0.26	14	<b>0.25</b>	14	0.42	<b>4</b>
mas74	0.58	28	0.66	28	0.60	28	<b>0.18</b>	<b>1</b>
mas76	0.53	28	0.68	28	0.59	28	<b>0.18</b>	<b>1</b>
mc11	1.96	23	2.31	23	2.12	23	<b>0.63</b>	<b>1</b>
mik-250-20-75-4	1.24	16	1.20	15	1.40	16	<b>0.64</b>	<b>3</b>
mushroom-best	40.92	15	54.92	15	25.36	11	<b>7.68</b>	<b>1</b>
n3div36	463.74	15	432.79	15	55.37	8	<b>42.86</b>	<b>3</b>
n5-3	1.34	15	1.50	15	1.64	13	<b>1.22</b>	<b>3</b>
n9-3	4.18	15	4.90	15	<b>2.79</b>	13	3.54	<b>2</b>
neos-1122047	35.61	15	38.03	15	46.73	15	<b>15.27</b>	<b>1</b>
neos-3083819-nubu	1.59	15	0.58	2	<b>0.42</b>	<b>1</b>	0.44	<b>1</b>
neos-4763324-toguru	<b>131.56</b>	23	145.39	23	298.17	22	798.40	<b>2</b>
neos-662469	65.61	15	70.40	14	17.10	14	<b>3.46</b>	<b>1</b>
neos-787933	70.45	16	69.52	16	60.03	16	<b>22.40</b>	<b>1</b>
neos-860300	42.63	15	43.74	14	42.61	14	<b>8.30</b>	<b>1</b>
neos-911970	0.89	17	1.05	17	0.96	17	<b>0.28</b>	<b>1</b>
neos-933966	9.60	15	14.28	15	5.00	<b>1</b>	<b>4.93</b>	<b>1</b>
neos17	2.00	16	2.39	16	2.54	15	<b>1.32</b>	<b>6</b>
neos5	0.47	15	0.61	15	0.40	13	<b>0.19</b>	<b>1</b>
nexp-150-20-8-5	<b>22.55</b>	15	24.78	15	28.09	<b>12</b>	73.01	21
ns1760995	878.32	15	878.51	15	762.52	12	<b>609.89</b>	<b>4</b>
nu25-pr12	2.93	15	3.67	15	<b>0.60</b>	<b>1</b>	<b>0.60</b>	<b>1</b>
p200x1188c	1.55	26	1.67	26	1.72	26	<b>0.41</b>	<b>1</b>

Continued on next page

TABLE 7. Runtimes and number of iterations for instances with positive RRF (all values are standard rounded)

Instances	ClassicBin		ScalingBin		maxScalingBin		PureScaling	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
pg	1.04	15	<b>0.22</b>	<b>1</b>	0.24	<b>1</b>	0.27	<b>1</b>
pk1	0.23	15	0.16	15	<b>0.11</b>	<b>1</b>	<b>0.11</b>	<b>1</b>
rai102	> 7200	-	> 7200	-	> 7200	-	<b>4280.51</b>	<b>1</b>
ran14x18-disj-8	<b>1.27</b>	15	2.14	15	1.56	14	1.80	<b>9</b>
reblock115	0.96	15	0.36	2	0.37	<b>1</b>	<b>0.35</b>	<b>1</b>
rococoB10-011000	7.53	22	3.92	21	2.25	22	<b>0.69</b>	<b>1</b>
rococoC11-011100	3.25	22	3.18	22	3.32	22	<b>1.03</b>	<b>1</b>
s250r10	217.94	15	171.28	3	<b>24.75</b>	<b>1</b>	24.96	<b>1</b>
satellites2-40	6.26	15	6.58	15	6.83	15	<b>2.95</b>	<b>1</b>
satellites2-60-fs	4.53	15	4.77	15	4.39	15	<b>1.64</b>	<b>1</b>
savsched1	124.35	15	180.82	15	<b>42.52</b>	<b>1</b>	42.60	<b>1</b>
seymour1	6.81	15	9.01	15	5.86	14	<b>1.45</b>	<b>1</b>
sp150x300d	0.44	24	0.51	24	0.40	24	<b>0.18</b>	<b>1</b>
square41	5800.64	15	6857.40	8	266.05	5	<b>74.89</b>	<b>1</b>
square47	> 7200	-	> 7200	-	743.13	5	<b>248.40</b>	<b>1</b>
supportcase18	<b>17.49</b>	18	18.61	17	334.40	15	239.92	<b>2</b>
supportcase42	<b>57.63</b>	15	114.86	15	105.69	<b>2</b>	76.04	<b>2</b>
swath1	<b>73.76</b>	19	74.16	19	2671.01	19	4622.70	<b>10</b>
trento1	<b>350.85</b>	<b>16</b>	354.43	<b>16</b>	1724.57	<b>16</b>	> 7200	-
var-smallemergy-m6j6	461.69	21	526.71	21	457.99	20	<b>203.29</b>	<b>7</b>

TABLE 8. Runtimes and number of iterations for instances with RRF zero (all values are standard rounded)

Instances	ClassicBin		ScalingBin		maxScalingBin		PureScaling	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
academictimetablesmall	69.20	16	69.23	16	67.34	1	<b>67.26</b>	<b>1</b>
app1-2	25.48	15	25.78	15	<b>20.14</b>	<b>1</b>	20.37	<b>1</b>
atlanta-ip	59.33	15	61.03	15	<b>8.07</b>	<b>1</b>	8.13	<b>1</b>
beasleyC3	1.06	21	1.06	21	<b>0.28</b>	<b>1</b>	0.30	<b>1</b>
bnatt400	564.01	16	564.76	16	158.92	<b>1</b>	<b>158.79</b>	<b>1</b>
cbs-cta	5.67	15	5.95	15	<b>1.15</b>	<b>1</b>	1.16	<b>1</b>
cmflsp50-24-8-8	73.25	15	73.38	15	<b>23.22</b>	<b>1</b>	23.25	<b>1</b>
co-100	211.70	15	215.46	15	<b>25.26</b>	<b>1</b>	26.20	<b>1</b>
dano3_3	<b>20.21</b>	15	20.58	15	79.85	<b>1</b>	80.25	<b>1</b>
dano3_5	<b>20.87</b>	15	21.03	15	161.70	<b>1</b>	162.33	<b>1</b>
decomp2	5.33	16	5.37	16	1.93	<b>1</b>	<b>1.91</b>	<b>1</b>
exp-1-500-5-5	0.40	24	0.38	24	0.17	<b>1</b>	<b>0.16</b>	<b>1</b>
fastxgemm-n2r6s0t2	3.83	15	3.96	15	<b>3.04</b>	<b>1</b>	3.05	<b>1</b>
germanrr	220.16	16	220.30	16	5.58	<b>1</b>	<b>5.49</b>	<b>1</b>
h80x6320d	5.30	23	5.06	23	<b>0.93</b>	<b>1</b>	0.98	<b>1</b>
hypothyroid-k1	<b>70.67</b>	16	73.34	16	94.91	<b>1</b>	94.67	<b>1</b>
irish-electricity	87.55	15	88.96	15	36.56	<b>1</b>	<b>36.32</b>	<b>1</b>
istanbul-no-cutoff	13.25	15	13.62	15	<b>4.68</b>	<b>1</b>	4.74	<b>1</b>
k1mushroom	<b>381.57</b>	19	396.44	19	504.87	<b>1</b>	505.16	<b>1</b>
lectsched-5-obj	70.85	15	71.73	15	<b>6.66</b>	<b>1</b>	6.77	<b>1</b>
lotsize	11.90	29	11.72	29	<b>0.35</b>	<b>1</b>	0.37	<b>1</b>
map10	332.48	15	335.44	15	<b>249.34</b>	<b>1</b>	250.87	<b>1</b>
map16715-04	336.61	15	337.06	15	250.81	<b>1</b>	<b>250.11</b>	<b>1</b>

Continued on next page

TABLE 8. Runtimes and number of iterations for instances with RRF zero (all values are standard rounded)

Instances	ClassicBin		ScalingBin		maxScalingBin		PureScaling	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
mcsched	0.26	15	0.32	15	<b>0.18</b>	<b>1</b>	0.19	<b>1</b>
milo-v12-6-r2-40-1	9.62	20	9.64	20	<b>2.65</b>	<b>1</b>	2.66	<b>1</b>
momentum1	92.91	15	93.54	15	<b>4.11</b>	<b>1</b>	4.25	<b>1</b>
mzzv11	1.62	15	1.57	15	1.43	<b>1</b>	<b>1.38</b>	<b>1</b>
mzzv42z	3.95	15	3.80	15	1.55	<b>1</b>	<b>1.53</b>	<b>1</b>
n2seq36q	1063.70	<b>16</b>	<b>1062.89</b>	<b>16</b>	> 7200	-	> 7200	-
neos-1171448	16.10	16	16.94	16	<b>2.08</b>	<b>1</b>	2.12	<b>1</b>
neos-1171737	8.33	17	8.66	17	<b>0.89</b>	<b>1</b>	0.92	<b>1</b>
neos-1354092	4447.14	15	4490.15	15	<b>504.62</b>	<b>1</b>	504.71	<b>1</b>
neos-1582420	<b>5.55</b>	15	5.60	15	20.31	15	16.59	<b>1</b>
neos-2657525-crna	1.43	17	1.47	17	<b>1.33</b>	<b>1</b>	1.33	<b>1</b>
neos-2978193-inde	<b>4.72</b>	18	4.80	18	5.50	<b>1</b>	5.44	<b>1</b>
neos-3216931-puriri	96.38	16	96.38	16	<b>17.27</b>	<b>1</b>	17.33	<b>1</b>
neos-3627168-kasai	1.20	27	1.21	27	<b>0.44</b>	<b>1</b>	0.45	<b>1</b>
neos-4413714-turia	83.27	16	89.39	16	15.44	<b>1</b>	<b>15.05</b>	<b>1</b>
neos-4532248-waihi	<b>817.56</b>	16	819.32	16	2224.22	16	2090.09	<b>1</b>
neos-4722843-widden	21.60	15	21.71	15	15.54	<b>1</b>	<b>14.99</b>	<b>1</b>
neos-4738912-atrato	3.41	18	3.32	18	<b>0.52</b>	<b>1</b>	0.54	<b>1</b>
neos-5049753-cuanza	<b>437.25</b>	15	444.66	15	529.72	<b>1</b>	529.12	<b>1</b>
neos-5075914-elvire	2526.50	<b>15</b>	<b>2526.25</b>	<b>15</b>	> 7200	-	> 7200	-
neos-5188808-nattai	23.46	15	24.03	15	<b>6.51</b>	<b>1</b>	<b>6.51</b>	<b>1</b>
neos-5195221-niemur	17.78	15	18.32	15	<b>11.96</b>	<b>1</b>	12.09	<b>1</b>
neos-631710	<b>46.16</b>	<b>17</b>	47.00	<b>17</b>	> 7200	-	> 7200	-

Continued on next page

TABLE 8. Runtimes and number of iterations for instances with RRF zero (all values are standard rounded)

Instances	ClassicBin		ScalingBin		maxScalingBin		PureScaling	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
neos-827175	<b>12.02</b>	15	12.46	15	20.32	<b>1</b>	20.27	<b>1</b>
neos-848589	270.69	35	277.71	35	<b>153.31</b>	<b>1</b>	153.33	<b>1</b>
neos-873061	378.84	15	380.52	15	226.97	<b>1</b>	<b>224.59</b>	<b>1</b>
neos-950242	9.93	15	9.85	15	<b>6.04</b>	<b>1</b>	<b>6.04</b>	<b>1</b>
neos-957323	18.75	15	19.15	15	<b>6.73</b>	<b>1</b>	6.91	<b>1</b>
neos-960392	<b>5.69</b>	15	5.76	15	19.06	<b>1</b>	19.09	<b>1</b>
neos8	23.48	15	24.16	15	9.39	<b>1</b>	<b>9.27</b>	<b>1</b>
net12	14.07	15	14.41	15	2.30	<b>1</b>	<b>2.22</b>	<b>1</b>
ns1116954	<b>1113.99</b>	15	1122.95	15	2565.66	<b>1</b>	2567.18	<b>1</b>
ns1208400	16.32	15	16.48	15	<b>16.11</b>	<b>1</b>	<b>16.11</b>	<b>1</b>
ns1644855	247.80	15	262.14	15	23.03	<b>1</b>	<b>22.90</b>	<b>1</b>
ns1830653	14.35	16	14.58	16	<b>1.89</b>	<b>1</b>	1.92	<b>1</b>
peg-solitaire-a3	157.90	16	<b>157.72</b>	16	246.30	<b>1</b>	246.12	<b>1</b>
pg5_34	1.82	20	1.84	20	<b>1.75</b>	<b>1</b>	1.76	<b>1</b>
physiciansched3-3	353.65	15	359.83	15	177.99	<b>1</b>	<b>175.61</b>	<b>1</b>
physiciansched6-2	76.86	15	77.16	15	<b>61.08</b>	<b>1</b>	61.61	<b>1</b>
piperout-08	6.13	15	6.21	15	<b>5.05</b>	<b>1</b>	5.13	<b>1</b>
radiation18-12-05	15.95	15	16.29	15	4.48	<b>1</b>	<b>4.31</b>	<b>1</b>
rail01	<b>461.10</b>	16	461.55	16	679.00	<b>1</b>	678.23	<b>1</b>
rd-rplusc-21	461.78	15	483.85	15	134.18	<b>1</b>	<b>132.79</b>	<b>1</b>
rocI-4-11	4.18	15	4.18	15	1.40	<b>1</b>	<b>1.39</b>	<b>1</b>
rocII-5-11	82.81	15	83.64	15	<b>6.58</b>	<b>1</b>	6.60	<b>1</b>
roi2alpha3n4	113.86	17	118.77	17	<b>10.26</b>	<b>1</b>	10.41	<b>1</b>

Continued on next page

TABLE 8. Runtimes and number of iterations for instances with RRF zero (all values are standard rounded)

Instances	ClassicBin		ScalingBin		maxScalingBin		PureScaling	
	time/s	niter	time/s	niter	time/s	niter	time/s	niter
roll3000	3.28	15	3.26	15	0.70	1	<b>0.68</b>	<b>1</b>
sct2	2.48	15	2.54	15	<b>0.50</b>	<b>1</b>	<b>0.50</b>	<b>1</b>
snp-02-004-104	13.46	15	13.77	15	<b>5.12</b>	<b>1</b>	5.18	<b>1</b>
sp97ar	32.29	15	33.98	15	<b>3.91</b>	<b>1</b>	4.00	<b>1</b>
sp98ar	47.00	15	51.23	15	<b>5.24</b>	<b>1</b>	5.29	<b>1</b>
supportcase12	102.61	15	105.57	15	29.87	<b>1</b>	<b>29.85</b>	<b>1</b>
supportcase33	18.95	15	19.27	15	<b>3.56</b>	<b>1</b>	3.73	<b>1</b>
supportcase40	17.96	15	19.10	15	<b>5.53</b>	<b>1</b>	5.62	<b>1</b>
supportcase7	302.96	15	318.74	15	29.36	<b>1</b>	<b>28.86</b>	<b>1</b>
tr12-30	<b>0.64</b>	25	0.68	25	0.94	<b>1</b>	0.94	<b>1</b>
traininstance6	3.22	15	3.22	15	<b>0.96</b>	<b>1</b>	1.01	<b>1</b>
triptim1	41.73	15	41.52	15	<b>19.92</b>	<b>1</b>	<b>19.92</b>	<b>1</b>
uccase12	71.50	15	75.47	15	<b>18.79</b>	<b>1</b>	18.97	<b>1</b>
uccase9	45.94	15	46.08	15	10.10	<b>1</b>	<b>10.07</b>	<b>1</b>
uct-subprob	<b>1.53</b>	15	1.59	15	3.37	<b>1</b>	3.35	<b>1</b>
unitcal_7	29.77	15	30.48	15	<b>5.11</b>	<b>1</b>	5.13	<b>1</b>