

Identifying relatively irreducible infeasible subsystems of linear inequalities

Jérémy Omer, Michael Poss

▶ **To cite this version:**

Jérémy Omer, Michael Poss. Identifying relatively irreducible infeasible subsystems of linear inequalities. Annals of Operations Research, 2021, 304 (1-2), pp.361-379. 10.1007/s10479-021-04126-9 . hal-03251527

HAL Id: hal-03251527
https://hal.science/hal-03251527

Submitted on 1 Sep 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

manuscript No.
(will be inserted by the editor)

Identifying relatively irreducible infeasible subsystems of linear inequalities

Jérémy Omer · Michael Poss

Received: date / Accepted: date

Abstract A classical problem in the study of an infeasible system of linear inequalities is to determine irreducible infeasible subsystems of inequalities (IISs), i.e., infeasible subsets of inequalities whose proper subsets are feasible. In this article, we examine a particular situation where only a given subsystem is of interest for the analysis of infeasibility. For this, we define relatively irreducible infeasible subsystems (RIISs) as infeasible subsystems of inequalities that are irreducible with respect to a given subsystem. It is a generalization of the definition of an IIS, since an IIS is irreducible with respect to the full system. We provide a practical characterization of RIISs, making the link with the alternative polyhedron commonly used in the detection of IISs. We then turn to the study of the RIISs that can be obtained from the *Phase I* of the simplex algorithm. We answer an open question regarding the covering of the clusters of IISs and show that this result cannot be generalized to RIISs. We thus develop a practical algorithm to find a covering of the clusters of RIISs. Our findings are numerically illustrated on the Netlib infeasible linear programs.

Keywords Systems of linear inequalities · Irreducible infeasible set · Conflict analysis · Linear programming

1 Introduction

When faced with a large system of inequalities, the knowledge that it is infeasible can be overwhelming if the analysis cannot be narrowed to smaller subsystems of infeasible inequalities. The best that can be done in this direction is identifying irreducible infeasible subsystems (IISs), i.e., infeasible subsystems whose proper subsystems are all feasible. The isolation of infeasibility has an obvious application

J. Omer
Univ Rennes, INSA, CNRS, IRMAR - UMR 6625, F-35000 Rennes, France
E-mail: jeremy.omer@insa-rennes.fr

M. Poss
LIRMM, University of Montpellier, CNRS
E-mail: michael.poss@lirmm.fr

in the diagnosis of infeasibility for practitioners who would like to understand why their model is infeasible.

IISs returned by large systems of inequalities typically contain many elements, making it hard to understand the source of infeasibility. It is thus essential to distinguish those inequalities that must be satisfied, because they describe the physics of a system, from those that can possibly be relaxed, for instance, involving some penalty or expansion cost. For example, transportation problems usually require flow conservation constraints and non-negativity constraints that cannot be excluded. In this case, the user is only interested in the constraints that are not intrinsic to the problem. Another example arises in supply chain problems where complex products are built through different steps to be ultimately shipped to clients. While the constraints describing the production can hardly be relaxed, the client demands are typically not hard constraints and they should, therefore, be the primary focus in understanding the system infeasibility.

The above observations led Chinneck and Dravnieks (1991) to split the problem constraints into, on the one hand, bounds and non-negativity constraints, and on the other hand, all other inequalities, called *functional constraints* therein. They then look for IISs that contain a minimal number of functional constraints. As finding minimum cardinality IISs is NP-hard (Amaldi et al. (2003)), so is the problem of finding IISs with few functional constraints. Chinneck (1997) thus presents different heuristic algorithms based on the deletion filtering method from Chinneck and Dravnieks (1991). This idea was further exploited by Chinneck (2008a), who details how *guide codes* can be used to enforce some constraints to hold. The notion of splitting constraints into those that must be enforced, and those that can possibly be relaxed has been used in other works on IISs, including Codato and Fischetti (2006) and Pfetsch (2008).

The purpose of this study is three-fold. First we wish to move one step beyond Chinneck (2008a); Chinneck and Dravnieks (1991) in formalizing the search of particular infeasible subsystems of inequalities when not every inequality is relevant to the analysis of infeasibility. Specifically, let $\mathcal{S}$ be an infeasible system of inequalities and assume that we wish to focus our analysis on some subsystem of $\mathcal{S}$, denoted $\mathcal{B}$: we define a relatively irreducible infeasible subsystem (denoted RIIS hereafter) as a subsystem of $\mathcal{S}$ that becomes feasible if any inequality of $\mathcal{B}$ is removed from it. Introducing RIISs formally allows to achieve minimality with respect to the subsystem of interest. Second, we detail the link between RIISs and the projection of the alternative polyhedron introduced in the study of IISs (e.g. in Pfetsch (2008)). We illustrate numerically on classical instances how working with the projection of this alternative polyhedron can be useful from a practical point of view. Last, we focus on the IISs and RIISs that can be obtained from the optimal solution of the Phase I of the simplex algorithm. We prove that these IISs cover all clusters of IISs, thus answering a conjecture of Chinneck and Dravnieks (1991). We exhibit a counter-example showing that this result does not generalize to RIISs. We thus develop an algorithm that computes a covering of the clusters of RIISs and illustate its execution on a classical benchmark.

The remainder of the paper is structured as follows. The next section recalls the definition of an IIS and its dual characterization. The concept of RIIS is then introduced in Section 3, and the alternative characterization is discussed in Section 4. Section 5 then turns to the Phase I of the simplex algorithm. Our

computational experiments are presented in Section 6 and concluding remarks are provided in Section 7.

1.1 Notations

In the remainder of the article, we will consider an infeasible system of m linear inequalities with unknown $x \in \mathbb{R}^n$:

$$\mathcal{S} : Ax \leq a, Bx \leq b,$$

where $A \in \mathbb{R}^{m_A \times n}$, $B \in \mathbb{R}^{m_B \times n}$, $a \in \mathbb{R}^{m_A}$, $b \in \mathbb{R}^{m_B}$ and $m_A + m_B = m$.

For a more concise presentation, we will also use the following notations.

- The systems $\{Ax \leq a\}$ and $\{Bx \leq b\}$ are respectively denoted $\mathcal{A}$ and $\mathcal{B}$.
- The matrix consisting of the rows of some matrix M indexed by a set I is denoted M_I and the vector consisting of the elements of some vector y indexed by a set I is denoted y_I.
- The support of a vector x is denoted as $\sigma(x)$.
- The subsystem of $\mathcal{B}$ that consists of the inequalities $B_J x \leq b_J$ is denoted as $\mathcal{B}_J$. By extension, the subsystem $\mathcal{A}_I \cup \mathcal{B}_J$ of $\mathcal{S}$ is $\{A_I x \leq a_I, B_J x \leq b_J\}$.

2 Irreducible infeasible subsystems of linear inequalities

We first formally define feasible and infeasible systems of linear inequalities, before giving the definition of an irreducible infeasible subsystem. The same notion can also be found under the names of *irreducibly inconsistent subsystem* (van Loon (1981)) and *minimally infeasible subsystem* (Gleeson and Ryan (1990)).

Definition 1 Let $\mathcal{S} : \{Ax \leq a, Bx \leq b\}$ be a system of m linear inequalities. System $\mathcal{S}$ is feasible if there exists x such that $Ax \leq a$ and $Bx \leq b$, and it is infeasible otherwise.

Definition 2 Let $\mathcal{S}$ be a system of linear inequalities. A subsystem $\mathcal{S}'$ of $\mathcal{S}$ is an irreducible infeasible subsystem of inequalities (IIS) if $\mathcal{S}'$ is infeasible, but every proper subsystem of $\mathcal{S}'$ is feasible.

It is well-known that IISs can be extracted from infeasible subsystems of linear inequalities using, for instance, the *filtering algorithm* of Chinneck and Dravnieks (1991), which iteratively constructs an IIS by removing constraints from an infeasible set. Thus, the identification of one IIS can be done in polynomial time. The inverse approach can also be followed to build an IIS in an additive algorithm (Tamiz et al. (1996)) that starts from the empty set and iteratively adds inequalities that trigger infeasibility. These two algorithms have been at the origin of a series of computational improvements reviewed by Chinneck (2008b). This has led to the implementation of filtering techniques in most commercial linear programming solvers (Chinneck (1997)).

In methods where the identification of IISs is necessary, but is not the final goal, it is in general more fruitful to rely on the following dual characterization of the set of IISs.

Theorem 1 (Gleeson and Ryan (1990)) *The indices of the IISs of $\mathcal{S}$ are in one-to-one correspondence with the supports of the vertices of the alternative polyhedron*

$$P := \left\{(y,z) \in \mathbb{R}^{m_A} \times \mathbb{R}^{m_B} \mid A^T y + B^T z = 0, a^T y + b^T z \leq -1, y \geq 0, z \geq 0\right\}$$

In particular, the nonzero components of any vertex of P index an IIS.

Sketch of proof *Farkas theorem of the Alternative shows that the support of any point of P describes an infeasible subsystem of $\mathcal{S}$. The second step of the proof uses the characterization of an extreme point to show that an infeasible subsystem of $\mathcal{S}$ is irreducible if and only if it corresponds to the support of a vertex of P.*

See the original publication by Gleeson and Ryan (1990) for the complete proof. □

A corollary of this characterization is that one IIS can be found by solving the linear program (LP)

$$\min\{c^T y + d^T z : \ A^T y + B^T z = 0, \ a^T y + b^T z \leq -1, \ y \geq 0, \ z \geq 0\},$$

where $c \in \mathbb{R}^{m_A}$and $d \in \mathbb{R}^{m_B}$. If c and d are set to nonnegative values the above LP has an optimal solution, and any extreme optimal solution is a vertex of P.

Fischetti et al. (2010) heuristically search for an IIS that includes a minimum number of disjunctive constraints by setting the corresponding dual costs to 1 for all such constraints and to 0 otherwise. In our formalism, the disjunctive constraints are given by $Bx \leq b$, so we would set $d = 1$ and $c = 0$. Another important application of the identification of IISs is the generation of the minimum cardinality set of constraints that need to be removed to recover the feasibility of the system. This problem is equivalent to a minimum weight IISs cover. To solve this problem, Parker and Ryan (1996) compute a minimum weight cover of a small set of IISs and then iteratively generate IISs that do not contain any inequality belonging to the cover. The generation of IISs is also carried out by solving a variant of the above LP.

3 Relatively irreducible infeasible subsystems

What we intend to accomplish is analyse the infeasibility of $\mathcal{S}$ with a focus on the role of the inequalities of $\mathcal{B}$. Specifically, we assume that inequalities of $\mathcal{A}$ hold and wish to understand how subsets of inequalities of $\mathcal{B}$ lead to infeasibilities. In the remainder of the article, we thus assume that $\mathcal{S}$ is infeasible, but $\mathcal{A}$ is feasible.

Definition 3 Let $\mathcal{B}_J$ be a subsystem of $\mathcal{B}$. We say that $\mathcal{B}_J$ is a relatively irreducible infeasible subsystem of inequalities (with respect to $\mathcal{B}$) if and only if it satisfies

$$\mathcal{A} \cup \mathcal{B}_J \text{ is infeasible and } \mathcal{A} \cup \mathcal{B}_{J'} \text{ is feasible, } \forall J' \subsetneq J \tag{1}$$

In the remainder of the article, we will call such subsystem an RIIS.

The above definition allows to generalize that of an IIS, because an IIS is actually irreducible with respect to the complete system $\mathcal{S}$. We illustrate in the the following example the relationship between IISs and RIISs.

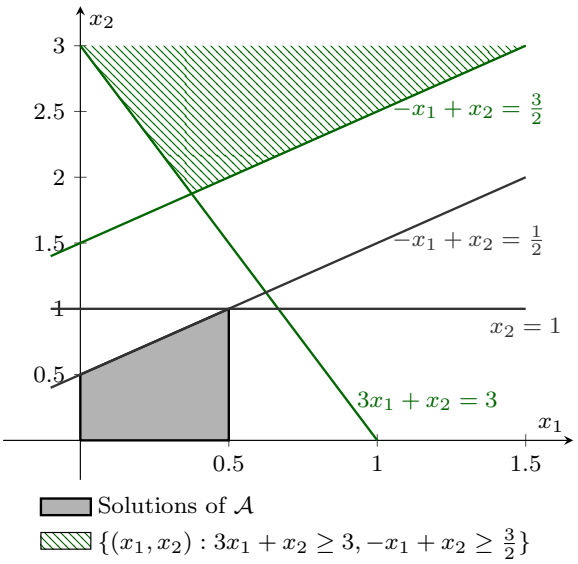


Fig. 1 Graphic representation of $\mathcal{A}$ and $\mathcal{B}$ (Example 1.)

Example 1 Let $\mathcal{S}$ be defined by the subsystems

$$\mathcal{A}: \begin{cases} -x_1 \leq 0 \\ x_1 \leq \dfrac{1}{2} \\ -x_2 \leq 0 \\ x_2 \leq 1 \\ -x_1 + x_2 \leq \dfrac{1}{2} \end{cases} \quad \text{and} \quad \mathcal{B}: \begin{cases} -3x_1 - x_2 \leq -3 \\ x_1 - x_2 \leq -\dfrac{3}{2} \end{cases}$$

A graphic representation of the inequalities of $\mathcal{A}$ and $\mathcal{B}$ appears in Figure 1.

The enumeration of the IISs of the complete system provides the following five sets of inequalities.

- $\{-x_1 + x_2 \leq \frac{1}{2}, -x_1 + x_2 \geq \frac{3}{2}\}$,
- $\{x_1 \geq 0, x_2 \leq 1, -x_1 + x_2 \geq \frac{3}{2}\}$,
- $\{x_1 \leq \frac{1}{2}, x_2 \leq 1, 3x_1 + x_2 \geq 3\}$,
- $\{x_1 \leq \frac{1}{2}, -x_1 + x_2 \leq \frac{1}{2}, 3x_1 + x_2 \geq 3\}$,
- $\{x_2 \leq 1, -x_1 + x_2 \geq \frac{3}{2}, 3x_1 + x_2 \geq 3, \}$.

In contrast, there are only two RIISs:

- $\{-x_1 + x_2 \geq \frac{3}{2}\}$, and
- $\{3x_1 + x_2 \geq 3\}$.

In this example, we observe that the last IIS of the list would not be as helpful as an RIIS in an analysis of infeasiblity that focuses on $\mathcal{B}$ since it contains the whole system $\mathcal{B}$. Also, the first two IISs would actually provide redundant information with respect to $\mathcal{B}$. □

Remark 1 By definition, for all RIIS, $\mathcal{B}_J$, there exists an IIS of $\mathcal{S}$, $\mathcal{A}_I \cup \mathcal{B}_J$. Conversely, there is no guarantee that $\mathcal{B}_J$ is an RIIS if $\mathcal{A}_I \cup \mathcal{B}_J$ is an IIS of $\mathcal{S}$. This is illustrated in the last IIS from Example 1.

For illustration, we treat several simple cases. Given that $\mathcal{S}$ is infeasible and $\mathcal{A}$ is feasible,

- if $\mathcal{B} = \emptyset$, then the feasibility of $\mathcal{A}$ is in contradiction with the infeasibility of $\mathcal{S}$.
- if $\mathcal{A} = \emptyset$, then the search for an RIIS comes down to the search of an IIS of $\mathcal{S}$.
- if $\mathcal{B}$ consists of a single inequality, then $\mathcal{B}$ is the only RIIS.

Moreover, in the case where $\mathcal{A}$ is infeasible, then there is no RIIS, because for every subsystem $\mathcal{B}_J$ of $\mathcal{B}$, $\{x \mid Ax \leq a, B_J x \leq b_J\} = \emptyset$. As a consequence, the second part of condition (1) can never be satisfied. In contrast, the case where $\mathcal{B}$ is infeasible does not exhibit particular properties in general. For instance, one can readily construct examples similar to the one above where IISs and RIISs yield different information.

4 Alternative polyhedron and the filtering method

Let $P := \left\{(y,z) \in \mathbb{R}^{m_A} \times \mathbb{R}^{m_B} \mid y^T A + z^T B = 0,\ y^T a + z^T b \leq -1,\ y \geq 0,\ z \geq 0\right\}$, be the alternative polyhedron that appears in the dual characterization of Theorem 1. We have seen that an extreme point of P can be found by minimizing $c^T y + d^T z$ subject to $(y,z) \in P$ for any nonnegative cost vectors c and d. Fischetti et al. (2010) and Codato and Fischetti (2006) suggest different values of c and d in heuristic algorithms that aim at producing IISs that include a minimum number of inequalities of $\mathcal{B}$. In particular, Codato and Fischetti (2006) set $c = 0$ and sample random nonnegative vectors d to generate several distinct IISs. In this section, we study how the dual characterization of IISs can be adapted to narrow the search for RIISs. This yields a clear connection between the heuristic algorithm of Codato and Fischetti (2006) and the extreme points of the projection of P on the space of z variables.

Given that we are interested in the support of variables z, it is natural to study the projection of P on the space of variables z,

$$\mathrm{Proj}_z(P) := \left\{z \geq 0 \mid \exists y \geq 0,\ y^T A + z^T B = 0,\ y^T a + z^T b \leq -1\right\}.$$

As a projection of P on a linear subspace, for all extreme points of $\mathrm{Proj}_z(P)$, $\hat{z}$, there is $\hat{y}$ such that $(\hat{y}, \hat{z})$ is an extreme point of P. However, it is not true that for all extreme point of P, $(\hat{y}, \hat{z})$, $\hat{z}$ is an extreme point of $\mathrm{Proj}_z(P)$. One can consider the projection of a pyramid on the plane of its base for a counterexample.

4.1 Link between the vertices of $\mathrm{Proj}_z(P)$ and RIISs

We investigate next whether Theorem 1 can be adapted to characterize RIISs. The following result shows that any RIIS corresponds to an extreme point of $\mathrm{Proj}_z(P)$.

Proposition 1 *Let $\mathcal{B}_J$ be an RIIS, then there is an extreme point of* $\mathrm{Proj}_z(P)$ *whose support coincides with J.*

Proof If $\mathcal{B}_J$ is an RIIS, there is $I \subseteq \{1,\dots,m_A\}$ such that $\mathcal{A}_I \cup \mathcal{B}_J$ is an IIS of $\mathcal{S}$. By Theorem 1, there is an extreme point $(\hat{y},\hat{z})$ of P whose support is given by $I \cup J$. Given that $\hat{z}$ is in $\mathrm{Proj}_z(P)$, it can be writen as a combination of $K \leq m_B+1$ extreme points, $z^1,\dots,z^K$, and one ray r of $\mathrm{Proj}_z(P)$:

$$\hat{z} = \sum_{k=1}^{K} \alpha_k z^k + \beta r,\ \sum_{k=1}^{K} \alpha_k = 1,\ 0 \leq \alpha_k \leq 1, k = 1,\dots,K,\ \beta \geq 0.$$

Let $k \in \{1,\dots,K\}$. Since $z \geq 0$ for all $z \in \mathrm{Proj}_z(P)$ and $r \geq 0$ for every ray of $\mathrm{Proj}_z(P)$, the support of z^k is necessarily included in that of $\hat{z}$, i.e., $\sigma(z^k) \subseteq J$. Besides, there is y^k such that (y^k, z^k) is an extreme point of P, so $\mathcal{A}_{\sigma(y^k)} \cup \mathcal{B}_{\sigma(z^k)}$ is an IIS of $\mathcal{S}$. Given that $\mathcal{B}_J$ is an RIIS of $\mathcal{S}$, we get that $J \subseteq \sigma(z^k)$, hence $\sigma(z^k) = J$.

As a conclusion, either $\hat{z}$ is an extreme point of $\mathrm{Proj}_z(P)$ with support equal to J, or any extreme point of $\mathrm{Proj}_z(P)$ among $z^1,\dots,z^K$ is supported by J. □

Proposition 1 guarantees that we can focus on the vertices of $\mathrm{Proj}_z(P)$ instead of those of P, since no RIIS will be left aside by doing so. Unfortunately, the converse of the proposition does not hold as there may exist a vertex of $\mathrm{Proj}_z(P)$ that is not supported by an RIIS. This is illustrated in the example below.

Example 2 Let $\mathcal{S}$ be defined by the subsystems

$$\mathcal{A}: \begin{cases} x_1 \leq 1 & [y_1] \\ x_2 \leq 1 & [y_2] \end{cases} \quad \text{and} \quad \mathcal{B}: \begin{cases} x_1 - x_2 \leq -\dfrac{1}{2} & [z_1] \\ -x_1 - x_2 \leq -3 & [z_2] \end{cases}$$

The alternative polyhedron is given by

$$P = \{(y,z) \geq 0 : y_1 + z_1 - z_2 = 0, y_2 - z_1 - z_2 = 0, y_1 + y_2 - \frac{1}{2} z_1 - 3z_2 \leq -1\}.$$

This polyhedron has two vertices, $(0, \frac{4}{3}, \frac{2}{3}, \frac{2}{3})$ and $(1,1,0,1)$ whose projections on the space of z variables are $(\frac{2}{3}, \frac{2}{3})$ and $(0,1)$. It so happens that $(\frac{2}{3}, \frac{2}{3})$ and $(0,1)$ are also the only vertices of $\mathrm{Proj}_z(P)$. However, there is only one RIIS that consists of the last constraint of $\mathcal{B}$: $\{-x_1 - x_2 \leq -3\}$. □

There are also cases where all vertices of $\mathrm{Proj}_z(P)$ lead to RIISs, as shown in the example below. The following example also illustrates that there are cases where $\mathcal{A}_I \cup \mathcal{B}_J$ is an IIS of $\mathcal{S}$, but no vertex of $\mathrm{Proj}_z(P)$ is supported by J. By Proposition 1, such subsystem $\mathcal{B}_J$ cannot be an RIIS, so we will need to apply some filtering algorithm to retrieve an RIIS from it. Given that this subsystem will not appear if we consider only the extreme points of $\mathrm{Proj}_z(P)$, this shows that there can be a benefit in focussing on the vertices of $\mathrm{Proj}_z(P)$ instead of those of P.

Example 3 We consider once again the system defined in Example 1.

$$\mathcal{A}: \begin{cases} -x_1 \leq 0 & [y_1] \\ x_1 \leq \frac{1}{2} & [y_2] \\ -x_2 \leq 0 & [y_3] \\ x_2 \leq 1 & [y_4] \\ -x_1 + x_2 \leq \frac{1}{2} & [y_5] \end{cases} \quad \text{and} \quad \mathcal{B}: \begin{cases} -3x_1 - x_2 \leq -3 & [z_1] \\ x_1 - x_2 \leq -\frac{3}{2} & [z_2] \end{cases}$$

The alternative polyhedron is given by

$$P = \left\{ \begin{array}{rl} (y,z) \in \mathbb{R}^5 \times \mathbb{R}^2 \ : & -y_1 + y_2 - y_5 - 3z_1 + z_2 = 0, \\ & -y_3 + y_4 + y_5 - z_1 - z_2 = 0, \\ & \frac{1}{2}y_2 + y_4 + \frac{1}{2}y_5 - 3z_1 - \frac{3}{2}z_2 \leq -1, \\ & y \geq 0, z \geq 0 \end{array} \right\}$$

Using the Polyhedra[1] and CDDLib[2] packages of the Julia language (Bezanson et al. (2017)), we enumerate the vertices of P as $(0,0,0,0,1,0,1)$, $(2,0,0,2,0,0,2)$, $(0,6,0,2,0,2,0)$, $(0,8,0,0,2,2,0)$ and $(0,0,0,\frac{8}{7},0,\frac{2}{7},\frac{6}{7})$, whose supports correspond exactly to the IISs enumerated in Example 1. The projection of these vertices on z variables yield three subsystems of $\mathcal{B}$: $\{-3x_1 - x_2 \leq -3\}$, $\{x_1 - x_2 \leq -\frac{3}{2}\}$ and $\{-3x_1 - x_2 \leq -3, x_1 - x_2 \leq -\frac{3}{2}\}$. In contrast, $\text{Proj}_z(P)$, has only two vertices, $(2,0)$ and $(0,1)$, whose supports correspond exactly to the RIISs of the system. □

4.2 Generalization of the filtering and additive methods

Let $\sigma_z(P)$ be the set of supports of the projection of the extreme points of P on the space of variables z, and let $\sigma(\text{Proj}_z(P))$ be the set of supports of the extreme points of $\text{Proj}_z(P)$. As discussed in the previous section, $\sigma(\text{Proj}_z(P)) \subseteq \sigma_z(P)$ and every RIIS is indexed by an element of $\sigma(\text{Proj}_z(P))$. Moreover, there are cases where $\sigma(\text{Proj}_z(P)) \subsetneq \sigma_z(P)$. As a consequence, focusing on $\text{Proj}_z(P)$ rather than P, we possibly dodge some extreme points that do not correspond to RIISs. This motivates the following result which shows that vertices of $\text{Proj}_z(P)$ can be computed just like those of P, i.e., by solving

$$\min\{c^T y + d^T z : \ A^T y + B^T z = 0, \ a^T y + b^T z \leq -1, \ y \geq 0, \ z \geq 0\}. \tag{2}$$

The only specificity is that c should be set to zero and d needs to be generated randomly from a vector of continuous random variables.

Proposition 2 *Let d be a vector of m_B mutually independent non-negative continuous random variables. Then, if $(\hat{y}, \hat{z}) \in \text{argmin}\{d^T z : (y,z) \in P\}$, $\hat{z}$ is a vertex of $\text{Proj}_z(P)$ with probability 1.*

[1] `https://github.com/JuliaPolyhedra/Polyhedra.jl`

[2] `https://github.com/JuliaPolyhedra/CDDLib.jl`

Proof Let $\bar{d} \in \mathbb{R}_+^{m_B}$ be a realization of d. Given that $P \neq \emptyset$ and $z \geq 0$ for all $(y, z) \in P$, then the linear program $\min\{\bar{d}^T z : (y, z) \in P\}$ is feasible and bounded. As a consequence, $\operatorname{argmin}\{\bar{d}^T z : (y, z) \in P\}$ is a nonempty face of P whose projection on the space of z is denoted as $Z_{\bar{d}}$. More formally,

$$Z_{\bar{d}} = \operatorname{Proj}_z\Big(\operatorname{argmin}\{\bar{d}^T z : (y, z) \in P\}\Big) \neq \emptyset.$$

Assume that $Z_{\bar{d}}$ is reduced to one point $\hat{z}$ (i.e., $\dim(Z_{\bar{d}}) = 0$). Given that $(\hat{y}, \hat{z}) \in P$, $\hat{z} \in \operatorname{Proj}_z(P)$. So, if $\hat{z}$ is not a vertex of $\operatorname{Proj}_z(P)$, there exist two distinct points z^1 and z^2 of $\operatorname{Proj}_z(P)$ such that $\hat{z} = \alpha z^1 + (1-\alpha) z^2$ for some $\alpha \in]0, 1[$. This is possible only if $\bar{d}^T z^1 = \bar{d}^T z^2 = \min\{\bar{d}^T z : (y, z) \in P\}$, a contradiction.

If $\dim(Z_{\bar{d}}) \geq 1$, then $Z_{\bar{d}}$ contains two distinct points z_1 and z_2 such that $\bar{d}^T(z_2 - z_1) = 0$. Since d is a vector of continuous random variables, for any two given vectors z_1, z_2 in $\operatorname{Proj}_z(P)$, the probability that $d^T(z_2 - z_1) = 0$ is zero. As a consequence, if we denote F a face of P whose projection on the space of z has nonzero dimension, then $\mathbb{P}\Big(\operatorname{argmin}\{d^T z : (y, z) \in P\} = F\Big) = 0$. Given that P has a finite number of faces, this yields

$$\mathbb{P}(\dim(Z_d) \geq 1) = 0.$$

We deduce that with probability 1, $\operatorname{argmin}\{d^T z : (y, z) \in P\}$ is reduced to one vertex of $\operatorname{Proj}_z(P)$. □

The significance of Proposition 2 is that it provides a practical method for computing a vertex of $\operatorname{Proj}_z(P)$. Indeed, when compared to the computation of a vertex of P, the only additional operation is the random generation of the cost vector d.

Remark 2 Tolerances in optimality and feasibility will necessarily yield a nonzero probability that the solution of $\min\{d^T z : (y, z) \in P\}$ be not a vertex of $\operatorname{Proj}_z(P)$. In practice though, if (2) is solved with a simplex or a crossover algorithm, we still obtain a vertex of P from which an RIIS can be found by filtering.

Once a vertex has been found, an RIIS can be found using, for instance, by the filtering algorithm described in Algorithm 1. A similar filtering algorithm has already been described by Chinneck (2008b), Section 6.2.6, where the constraints can be labeled to guide the isolation of infeasibility. An RIIS can be otained with this code by protecting the inequalities of $\mathcal{A}$ from elimination with the relevant label. The specificity of Algorithm 1 is that we start the filtering algorithm from the support of a vertex $\operatorname{Proj}_z(P)$. We also show that it guarantees the identification of an RIIS in Proposition 3.

Proposition 3 *Let $\mathcal{B}_J$ be a subsystem of $\mathcal{B}$ such that $\{x \mid Ax \leq a, B_J x \leq b_J\} = \emptyset$, then it is possible to extract at least one RIIS, $J' \subseteq J$.*

Proof We show the result by induction on the size of $\mathcal{B}_J$. If $\mathcal{B}_J$ consists of only one inequality, then it is an RIIS. Now, assume that $\mathcal{B}_J$ includes more than one constraint. If there exists $J' \subsetneq J$ such that $\{x \mid Ax \leq a, B_{J'} x \leq b_{J'}\} = \emptyset$, then apply the induction hypothesis on $\mathcal{B}_{J'}$. Otherwise, $\mathcal{B}_J$ is an RIIS (by definition).
□

Input: The systems $\mathcal{A} : Ax \leq a$ and $\mathcal{B} : Bx \leq b$;
Solve (2) for random c and d // $c = 0$ `if we seek an extreme point of` $\text{Proj}_z(P)$
Let (y^*, z^*) be an optimal extreme solution and $Z \leftarrow \sigma(z^*)$;
$J \leftarrow Z$;
for $j \in Z$ **do**
  if $j \in J$ **then**
    $J' \leftarrow J \setminus \{j\}$;
    Solve
     $(\text{LP}_{J'}) \leftarrow \min\{c^T y + d_{J'}^T z : A^T y + B_{J'}^T z = 0,\, a^T y + b_{J'}^T z \leq -1,\, y \geq 0,\, z \geq 0\}$;
    if $(\text{LP}_{J'})$ *is feasible* // `i.e.,` $\mathcal{A} \cup \mathcal{B}_{J'}$ `is infeasible`
     then
      $(y', z') \leftarrow$ an optimal extreme solution of $(\text{LP}_{J'})$;
      $J \leftarrow \sigma(z')$ // j `is excluded from` J `and we perform an opportunistic removal of other elements from` J `if` $\sigma(z') \subsetneq J'$
return J;

Algorithm 1: Filtering algorithm to find an RIIS

A similar result is also valid for the generalization of the additive methods of Tamiz et al. (1996). Since the feasibility of a system of linear inequalities can be verified in polynomial time, this is also true for the extraction of one RIIS from a subsystem of $\mathcal{B}$.

The next section turns to the analysis of the IISs and RIISs that can be deduced from the solution of the Phase I LP.

5 Phase I sensitivity analysis

In the search for an initial feasible solution for the primal simplex algorithm, it is classical to solve a so-called *Phase I LP* (see e.g. Dantzig and Thapa (1997) for a complete description). The idea is to add one non-negative artificial variable to each constraint so that a trivial feasible solution appears, and minimize the sum of these variables. A nonzero optimal value then means that the LP is infeasible. Chinneck and Dravnieks (1991) discuss how this applies to the search for an IIS of the system $\{x \geq 0, Bx \leq b\}$, where $Bx \leq b$ are called *functional constraints* in opposition to the nonnegativity constraints. This formalism is a specific case of ours, where $\mathcal{A} : x \leq 0$ and $\mathcal{B} : Bx \leq b$. In their work, Chinneck and Dravnieks (1991) introduce the concept of *irreducible inconsistent set of functional constraints* (IISF) as the complete subset of functional constraints involved in an IIS. However, they do not investigate the minimality of such IISF.

Taking the more general framework of this article, where $\mathcal{A} : Ax \leq a$ can be any feasible system of linear inequalities, the Phase I LP considered by Chinneck and Dravnieks (1991) is

$$\text{LP}_\text{I} : \begin{cases} \min & \mathbf{1}^T s & \\ \text{subject to} & Ax \leq a, & [y] \\ & Bx - s \leq b, & [z] \\ & s \geq 0 & \end{cases}$$

where **1** is a vector of ones with the appropriate dimension. The dual LP of the above can be equivalently written as

$$\mathrm{LD_I}: \begin{cases} -\min & a^T y + b^T z \\ \text{subject to} & A^T y + B^T z = 0, \\ & y \geq 0, 0 \leq z \leq \mathbf{1} \end{cases}$$

Given that $\mathcal{A}$ is feasible, $\mathrm{LP_I}$ and $\mathrm{LD_I}$ both have optimal solutions. In the remainder of this section, (x^*, s^*) is an extreme optimal solution of $\mathrm{LP_I}$ and (y^*, z^*) is a complementary extreme optimal solution of $\mathrm{LD_I}$. Chinneck and Dravnieks (1991) show several properties for the system $\{x \geq 0, Bx \leq b\}$ that straightforwardly generalize as follows.

Property 1

1. $\sigma(s^*)$ is an RIIS cover, i.e., for all RIIS J, $\sigma(s^*) \cap J \neq \emptyset$;
2. $s_j^* > 0$ only if $B_j x \leq b_j$ belongs to an IIS;
3. $y_i^* > 0$ only if $A_i x \leq a_i$ belongs to an IIS;
4. if $J = \sigma(z^*)$, then J contains an RIIS.

One related observation is that the infeasibility analysis of a system may be simplified when the set of IISs can be partitioned into independent subsets called *clusters*. More precisely, the clusters are the minimal sets of IISs such that two IISs sharing at least one constraint belong to the same cluster. In what follows, we establish in Theorem 2 that the support of (y^*, z^*) contains the indices of at least one IIS from each cluster, as conjectured by Chinneck and Dravnieks (1991). We then generalize the definition of a cluster to RIISs, and we exhibit a counter-example showing that there can be a cluster of RIISs containing no RIIS whose indices are included in the support of z^*.

Lemma 1 *Let $l \in \{1, \ldots, m_B\}$ be such that $s_l^* > 0$. There exists at least one IIS of $\mathcal{S}$, $\mathcal{A}_I \cup \mathcal{B}_J$, such that $l \in J$, $y_i^* > 0, \forall i \in I$, and $z_j^* > 0, \forall j \in J$.*

Proof By complementarity of the primal and dual solutions, we know that for all $l \in \sigma(s^*)$, $l \in \sigma(z^*)$. Let $\overline{\mathrm{LP_I}}$ be the LP obtained from $\mathrm{LP_I}$ by keeping only the constraints of $\mathcal{A}$ indexed by $\sigma(y^*)$ and those of $\mathcal{B}$ indexed by $\sigma(z^*)$. Denoting $\bar{s} = s^*_{\sigma(z^*)}$, $\bar{y} = y^*_{\sigma(y^*)}$ and $\bar{z} = z^*_{\sigma(z^*)}$, one can readily verify that $(x^*, \bar{s})$ is an optimal solution of $\overline{\mathrm{LP_I}}$, by complementarity with the dual solution $(\bar{y}, \bar{z})$.

Now, let $l \in \{1, \ldots, m_B\}$ such that $s_l^* > 0$. Given that $\bar{s} = s^*_{\sigma(z^*)}$ and $l \in \sigma(z^*)$, the application of item 2 of Property 1 to $\overline{\mathrm{LP_I}}$ guarantees that $B_l x \leq b$ belongs to an IIS, $\mathcal{A}_I \cup \mathcal{B}_J$, of the constraints of $\overline{\mathrm{LP_I}}$. By definition of $\overline{\mathrm{LP_I}}$, we necessarily have $I \subseteq \sigma(y^*)$ and $J \subseteq \sigma(z^*)$. □

Theorem 2 *The support of (y^*, z^*), $\sigma(y^*) \cup \sigma(z^*)$, contains the indices of at least one IIS from each cluster of IISs.*

Proof Let C be a cluster of IISs and $I \cup J \in C$. From item 1 of Property 1, we know that there is $l \in \{1, \ldots, m_B\}$ such that $s_l^* > 0$ and $l \in J$. Now, from Lemma 1, there is an IIS, $\mathcal{A}_{I'} \cup \mathcal{B}_{J'}$, such that $I' \subseteq \sigma(y^*)$, $J' \subseteq \sigma(z^*)$ and $l \in J'$. Since l indexes an inequality involved in at least one IIS of C, $\mathcal{A}_{I'} \cup \mathcal{B}_{J'}$ must also belong to C, which concludes the proof. □

We now wish to see how this result generalizes to RIISs. For this we start by defining clusters of RIISs.

Definition 4 A set C of RIISs of $\mathcal{S}$ is a *cluster of RIISs* if and only if:

- $C \neq \emptyset$;
- if $\mathcal{B}_J$ and $\mathcal{B}_{J'}$ are two RIISs and $\mathcal{B}_J \in C$, then

$$(\mathcal{B}_J \cap \mathcal{B}_{J'} \neq \emptyset) \implies (\mathcal{B}_J \in C)\,.$$

The motivation for considering clusters of RIISs is that they can be more numerous than clusters of IISs. Indeed, two RIISs J_1 and J_2 can belong to different clusters of RIISs even though every pair of IISs, $\mathcal{A}_{I_1} \cup \mathcal{B}_{J_1}$ and $\mathcal{A}_{I_2} \cup \mathcal{B}_{J_2}$, belong to the same cluster of IISs.

Remark 3 One consequence of Theorem 2 is that for each cluster C of IISs there is an RIIS that is included in an IIS of C and in $\sigma(z^*)$. Indeed, this RIIS may be extracted from any IIS of C that is included in $\sigma(y^*) \cup \sigma(z^*)$.

Unfortunately, Theorem 2 does not generalize to RIISs. The following example shows there there is no guarantee that for each cluster of RIISs there is one RIIS included in $\sigma(z^*)$.

Example 4 Consider a system $\mathcal{S}$ defined by the following two subsystems.

$$\mathcal{A}: \begin{cases} -x_1 \leq 0 & [y_1] \\ -x_2 \leq 0 & [y_2] \end{cases} \quad \text{and} \quad \mathcal{B}: \begin{cases} -x_1 - x_2 \leq -2 & [z_1] \\ x_1 + x_2 \leq -2 & [z_2] \\ x_1 + x_2 \leq 1 & [z_3] \end{cases}$$

The systems contains two RIIS: $\{-x_1 - x_2 \leq -2, x_1 + x_2 \leq 1\}$ and $\{x_1 + x_2 \leq -2\}$. Given that the RIISs have empty intersections, they define two clusters of RIISs. Now, the corresponding Phase I LP is:

$$\mathrm{LP_I}: \begin{cases} \min & \mathbf{1}^T s & \\ \text{subject to} & -x_1 - x_2 - s_1 \leq -2 & [z_1] \\ & x_1 + x_2 - s_2 \leq -2 & [z_2] \\ & x_1 + x_2 - s_3 \leq 1 & [z_3] \\ & x, s \geq 0 & \end{cases}$$

Solving $\mathrm{LP_I}$ with a simplex algorithm, we get the extreme optimal solution $x^* = (0, 0)$ and $s^* = (2, 2, 0)$ with complementary dual solution $z^* = (1, -1, 0)$. We see that the indices of the RIIS $\{-x_1 - x_2 \leq -2, x_1 + x_2 \leq 1\}$ are not included in $\sigma(z^*)$, so no RIIS of the corresponding cluster is included in $\sigma(z^*)$.

In the following result, we specify how the optimal dual vector z^* can be decomposed as the combination of extreme points of $\mathrm{Proj}_z(P)$ and an additional vector $\bar{z}$ whose support indexes a feasible subsystem. The result and its constructive proof yield an algorithm that can produce several RIISs (at least one per cluster of IISs) from the solution of one Phase I LP.

Theorem 3 *The projection of the optimal dual solution on z variables, z^*, can be decomposed as*

$$z^* = \sum_{k=1}^{K} \alpha_k z^k + \bar{z}, \tag{3}$$

where $K \leq |\sigma(z^*)|$ and, for all $k \in \{1, \ldots, K\}$,

- $\alpha_k > 0$,
- z^k *is an extreme point of* $\mathrm{Proj}_z(P)$,
- $\mathcal{B}_{\sigma(z^k)}$ *is an RIIS,*
- *there is $j \in \sigma(z^k)$ such that $s^*_j > 0$.*

Moreover, $\mathcal{A} \cup \mathcal{B}_{\sigma(\bar{z})}$ is a feasible subsystem of $\mathcal{S}$.

Proof Let $\bar{z} := z^*$. From the assumption that $\mathcal{S}$ is not feasible, we know that $a^T y^* + b^T z^* = a^T y^* + b^T \bar{z} < 0$. As a consequence, $\mathcal{A}_{\sigma(y^*)} \cup \mathcal{B}_{\sigma(\bar{z})}$ is an infeasible subsystem of $\mathcal{S}$, and so is $\mathcal{A} \cup \mathcal{B}_{\sigma(\bar{z})}$. This implies that there is an RIIS $\mathcal{B}_J$ such that $J \subseteq \sigma(\bar{z})$.

By Proposition 1, we know that there is an extreme point z^1 of $\mathrm{Proj}_z(P)$ such that $\sigma(z^1) = J$. First, (x^*, s^*) is an optimal solution of $\mathrm{LP_I}$ so $Ax^* \leq a$ and $Bx^* - s^* \leq b$. In particular, we have $Ax^* \leq a$ and $B_{\sigma(z^1)}x^* - s^*_{\sigma(z^1)} \leq b_{\sigma(z^1)}$. But, $\mathcal{A} \cup \mathcal{B}_{\sigma(z^1)}$ is an infeasible subsytem of $\mathcal{S}$, so there must be $j \in \sigma(z^1)$ such that $s^*_j > 0$. Second, we have $\sigma(z^1) \subseteq \sigma(\bar{z})$ so there is $\alpha_1 > 0$ such that

$$\begin{aligned} &\sigma(\bar{z} - \alpha_1 z^1) \subsetneq \sigma(\bar{z}), \\ &\bar{z} - \alpha_1 z^1 \geq 0 \end{aligned}$$

Updating $\bar{z}$ as $\bar{z} \leftarrow \bar{z} - \alpha_1 z^1$, we can apply the above process recursively until $\mathcal{A} \cup \mathcal{B}_{\sigma(\bar{z})}$ is feasible. This yields the required decomposition in K steps. Moreover, the cardinality of the support of $\bar{z}$ decreases at each step of the recursion so $K \leq |\sigma(z^*)|$. □

The decomposition given in Theorem 3 and the recursion used in its proof provide a method for identifying a set of RIISs as described in Remark 3. Setting $\mathcal{A} = \emptyset$ and $\mathcal{B} = \mathcal{S}$, the decomposition can also provide at least one IIS per cluster

of IISs, as guaranteed by Theorem 2. To build this decomposition in practice, we can follow the algorithm below.

Input: A system $\mathcal{S} = \mathcal{A} \cup \mathcal{B}$;
$(x^*, s^*) \leftarrow$ an optimal extreme solution of $\mathrm{LP_I}$;
$(y^*, \bar{z}) \leftarrow$ a complementary dual solution;
$Z \leftarrow \sigma(\bar{z})$, $\mathcal{J} \leftarrow \emptyset$, $k \leftarrow 0$;
while $\mathcal{A} \cup \mathcal{B}_Z$ *is infeasible* **do**
  $k \leftarrow k + 1$;
  `// get one one RIIS from` Z
  $J \leftarrow$ extract an RIIS from $\mathcal{A} \cup \mathcal{B}_Z$ with Algorithm 1;
  $\mathcal{J} \leftarrow \mathcal{J} \cup \{\mathcal{B}_J\}$;
  `// get a vertex of` $\mathrm{Proj}_z(P)$
  Sample $\{d_j\}_{j \in J}$ from i.i.d. positive continuous distributions;
  $(y^k, z^k) \leftarrow$ an optimal extreme solution of
  $\min\{d_J^T z_J : A^T y + B_J^T z_J = 0,\ a^T y + b_J^T z_J = -1,\ y \geq 0,\ z_J \geq 0\}$;
  `// update` $\bar{z}$
  $\alpha_k := \min_{j \in J} \left\{ \frac{\bar{z}_j}{z_j^k} \right\}$;
  $\bar{z} \leftarrow \bar{z} - \alpha_k z^k$;
  $Z \leftarrow \sigma(\bar{z})$;
return $\mathcal{J}$;

Algorithm 2: Identification of RIISs using Phase I sensitivity analysis

Corollary 1 *At the end of Algorithm 2, for each cluster C of IISs there is at least one IIS $\mathcal{A}_I \cup \mathcal{B}_J \in C$ such that $\mathcal{B}_J \in \mathcal{J}$.*

Proof First observe that Algorithm 2 follows exactly the recursive process described in the proof of Theorem 3. At each step, Algorithm 1 is executed to get an RIIS whose support is included in $\sigma(\bar{z})$, and an LP is solved to get a dual solution (y^k, z^k) such that $\sigma(z^k) \subseteq \sigma(\bar{z})$. Moreover, by Proposition 2, z^k is an extreme point of $\mathrm{Proj}_z(P)$. Notice also that $\sigma(z^k) = J$ as $\mathcal{B}_J$ is an RIIS. Therefore, one readily verifies that the choice of α_k guarantees that

$$\sigma(\bar{z} - \alpha_k z^k) \subsetneq \sigma(\bar{z}),$$
$$\bar{z} - \alpha_k z^k \geq 0.$$

As a consequence, Algorithm 2 yields

$$z^* = \sum_{k=1}^{K} \alpha_k z^k + \bar{z},$$

where for all $k \in \{1, \ldots, K\}$, $\alpha_k > 0$ and $\mathcal{B}_{\sigma(z^k)}$ is an RIIS, and $\mathcal{A} \cup \mathcal{B}_{\sigma(\bar{z})}$ is a feasible subsystem of $\mathcal{S}$.

Now, let C be a cluster of IISs. By Theorem 2, there is $\mathcal{A}_I \cup \mathcal{B}_J \in C$ such that $J \subseteq \sigma(z^*)$. Given that $\mathcal{A} \cup \mathcal{B}_{\sigma(\bar{z})}$ is feasible, there must be $j \in J$ such that $j \in \sigma(z^*) \setminus \sigma(\bar{z})$. By definition of the decomposition, this means that there is $k \in \{1, \ldots, K\}$ such that $j \in \sigma(z^k)$, hence $J \cap \sigma(z^k) \neq \emptyset$. Since $\sigma(z^k)$ is an RIIS, this means that $\sigma(z^k) \in C$. □

Stated otherwise, Corollary 1 guarantees that Algorithm 2 provides at least as many independent RIISs as the number of clusters of IISs. This is not sufficient though to get one RIIS per cluster of RIISs. For this, we must iteratively call Algorithm 2 after removing the supports of the returned RIISs until none can be found. This is detailed in Algorithm 3.

Input: A system $\mathcal{S} = \mathcal{A} \cup \mathcal{B}$
$\mathcal{J} \leftarrow$ result of Algorithm 2 with input $\mathcal{S} = \mathcal{A} \cup \mathcal{B}$;
$\mathcal{J}_{\text{RIIS}} \leftarrow \emptyset$;
$\bar{J} \leftarrow \{1, \ldots, m_B\}$;
while $\mathcal{J} \neq \emptyset$ **do**
  $\mathcal{J}_{\text{RIIS}} \leftarrow \mathcal{J}_{\text{RIIS}} \cup \mathcal{J}$;
  `// remove covered inequalities of` $\mathcal{B}$
  $\bar{J} \leftarrow \bar{J} \setminus \left(\bigcup_{B_J \in \mathcal{J}} J\right)$;
  `// run another Phase I analysis`
  $\mathcal{J} \leftarrow$ result of Algorithm 2 with input $\mathcal{S} = \mathcal{A} \cup \mathcal{B}_{\bar{J}}$;
return $\mathcal{J}_{\text{RIIS}}$;

Algorithm 3: Identification of at least one RIIS per cluster of RIISs

Proposition 4 *At the end of Algorithm 3, for each cluster C of RIISs there is at least one RIIS $\mathcal{B}_J \in \mathcal{J}_{RIIS}$ such that $\mathcal{B}_J \in C$.*

Proof Observe that at the end of Algorithm 3,

$$\bar{J} = \{1, \ldots, m_B\} \setminus \left(\bigcup_{B_J \in \mathcal{J}_{\text{RIIS}}} J \right).$$

So, if a cluster of RIISs is not covered by $\mathcal{J}_{\text{RIIS}}$, it must include only RIISs whose indices belong to $\bar{J}$. But, Algorithm 3 ends when Algorithm 2 cannot extract any RIIS from $\mathcal{B}_{\bar{J}}$. Hence, $\mathcal{J}_{\text{RIIS}}$ contains at least one RIIS per cluster of RIISs.

6 Computational experiments

We detail below the results of our experiments realized on infeasible instances made available by Csaba Mészárosat[3], mentioned in Chinneck (2008b), among others. As these instances consider classical IISs, instead of RIISs, we need to create a decomposition of the linear systems involved into subsystems $\mathcal{A}$ and $\mathcal{B}$. For this, we consider that the inequalities that are most likely to represent structural constraints are the variable bounds and the inequalities with right-hand-side equal to 0. As a consequence, we initialize $\mathcal{A}$ with a feasible set of variable bounds. In most cases, this is just the set of all variable bounds, but when a set of bounds is inconsistent, we solve a Phase I problem and keep only the bound inequalities with a zero artificial slack variable. If it does not make $\mathcal{A}$ infeasible, we then add all the equalities with a zero right-hand-side. Likewise, we finally add to $\mathcal{A}$ all the inequalities with a zero right-hand-side if the system remains feasible after this.

[3] Every instance can be downloaded at `http://old.sztaki.hu/~meszaros/public_ftp/lptestset/infeas/`

name	#var	#ctr	#ctrA	#ctrB	|IIS|	$\|\sigma(z)\|$	|RIIS|	reduction (%)
bgdbg1	407	800	526	274	5.0	1.4	1.4	0.0
bgetam	688	1305	1280	25	20.7	3.1	1.0	67.7
bgindy	10116	12787	10713	2074	154.0	3.0	3.0	0.0
bgprtr	34	54	42	12	12.3	6.1	5.0	18.0
box1	261	492	261	231	9.3	8.3	8.0	3.6
ceria3d	824	3576	3381	195	149.9	8.9	4.0	55.1
chemcom	720	1152	1114	38	36.5	4.0	2.0	50.0
ex72a	215	412	215	197	59.2	58.2	58.2	0.0
ex73a	211	404	211	193	25.3	24.3	24.0	1.2
forest6	95	166	160	6	94.0	6.0	3.0	50.0
galenet	8	24	20	4	6.0	2.0	1.0	50.0
gams30am	181	714	362	352	61.0	10.5	1.0	90.5
gams60am	361	1434	722	712	121.0	21.8	1.0	95.4
gosh	10733	14283	14234	49	9.0	1.0	1.0	0.0
greenbea	5405	8200	5878	2322	55.0	7.2	1.0	86.1
itest2	4	13	6	7	3.0	1.0	1.0	0.0
itest6	8	19	11	8	3.8	2.7	1.2	55.6
klein1	54	108	104	4	55.0	4.0	4.0	0.0
klein2	54	531	517	14	55.0	3.0	1.0	66.7
klein3	88	1082	89	993	89.0	87.0	78.4	9.9
mondou2	604	1520	1332	188	47.8	24.4	21.0	13.9
pang	460	862	817	45	35.0	2.0	2.0	0.0
pilot4i	1000	1599	1436	163	198.3	20.7	1.0	95.2
qual	464	1037	931	106	241.9	17.0	8.0	52.9
reactor	637	1505	1199	306	9.2	1.0	1.0	0.0
vol1	464	1037	931	106	235.1	14.5	8.0	44.8
woodinfe	89	138	103	35	2.0	1.0	1.0	0.0
								33.6

Table 1 Instance characteristics, averages sizes of IISs and RIISs.

All other inequalities and equalities form the system $\mathcal{B}$, where each equality is replaced by two inequalities. The resulting instances are described in Table 1.

We used the dual simplex of Gurobi solver[4] to get extreme solutions of all the linear programs solved in Algorithm 1 and 2. All our algorithms are implemented with the Julia language. The code and the instances are publicly available on a git repository[5].

The purpose of our numerical experiments is three-fold. First, we illustrate the size of the RIISs we obtain for each instance with Algorithm 1, comparing these with the IISs we obtain by solving the dual problem (2). Second, we assess the interest of working with the projected polyhedron. For this, we set $c = 0$ and sample d randomly when searching for an RIIS. Third, we exemplify the clusters of RIISs obtained with Algorithm 3.

The main motivation behind the introduction of RIISs is the size of the infeasible systems that need to be analyzed, often manually, by the modelers and decision makers. To study this, we generate 100 IISs by solving (2) with 100 different random cost functions sampled independently (not necessarily focussing on the projection, e.g., c may be different from 0). We then extract one RIIS from each IIS with the filtering algorithm described in Algorithm 1. Table 1 reports the

[4] See `https://www.gurobi.com`

[5] See `https://gitlab.insa-rennes.fr/Jeremy.Omer/RIIS_public.git`. Notice that these instances are actually a superset of the original netlib set available at `http://www.netlib.org/lp/infeas/` where a couple of additional files have been added.

	$\lvert\sigma(z)\rvert$		|RIIS|		#it		reductions (%)		
name	P	$\text{Proj}_z(P)$	P	$\text{Proj}_z(P)$	P	$\text{Proj}_z(P)$	$\lvert\sigma(z)\rvert$	|RIIS|	#it
bgdbg1	1.4	1.3	1.4	1.3	1.4	1.3	7.1	7.1	7.1
bgetam	3.1	3.0	1.0	1.0	3.0	3.0	3.2	0.0	0.0
bgindy	3.0	3.0	3.0	3.0	3.0	3.0	0.0	0.0	0.0
bgprtr	6.1	6.7	5.0	5.0	6.0	6.2	−9.8	0.0	−3.3
box1	8.3	8.0	8.0	8.0	8.2	8.0	3.6	0.0	2.4
ceria3d	8.9	5.2	4.0	4.2	5.3	4.4	41.6	−5.0	17.0
chemcom	4.0	3.6	2.0	2.0	4.0	3.6	10.0	0.0	10.0
ex72a	58.2	58.2	58.2	58.2	58.2	58.2	0.0	0.0	0.0
ex73a	24.3	24.0	24.0	24.0	24.2	24.0	1.2	0.0	0.8
forest6	6.0	6.0	3.0	3.0	6.0	6.0	0.0	0.0	0.0
galenet	2.0	1.7	1.0	1.0	2.0	1.7	15.0	0.0	15.0
gams30am	10.5	1.0	1.0	1.0	2.0	1.0	90.5	0.0	50.0
gams60am	21.8	1.0	1.0	1.0	2.0	1.0	95.4	0.0	50.0
gosh	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0
greenbea	7.2	1.0	1.0	1.0	2.5	1.0	86.1	0.0	60.0
itest2	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0
itest6	2.7	1.6	1.2	1.4	2.0	1.5	40.7	−16.7	25.0
klein1	4.0	4.0	4.0	4.0	4.0	4.0	0.0	0.0	0.0
klein2	3.0	1.0	1.0	1.0	2.1	1.0	66.7	0.0	52.4
klein3	87.0	83.9	78.4	76.2	81.7	79.9	3.6	2.8	2.2
mondou2	24.4	22.8	21.0	20.3	23.6	22.6	6.6	3.3	4.2
pang	2.0	2.0	2.0	2.0	2.0	2.0	0.0	0.0	0.0
pilot4i	20.7	1.6	1.0	1.0	2.6	1.6	92.3	0.0	38.5
qual	17.0	16.5	8.0	8.0	14.0	13.8	2.9	0.0	1.4
reactor	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0
vol1	14.5	15.2	8.0	8.0	13.3	13.7	−4.8	0.0	−3.0
woodinfe	1.0	1.0	1.0	1.0	1.0	1.0	0.0	0.0	0.0
average							20.4	−0.3	12.2

Table 2 Obtaining RIISs from extreme points of P or $\text{Proj}_z(P)$.

average number of constraints returned in IISs (|IIS|), constraints of the IIS that belong to $\mathcal{B}$ ($|\sigma(z)|$), and constraints in RIISs (|RIIS|). The last column illustrates the effect of Algorithm 1, that is, the relative reduction obtained by filtering the constraints returned in $\sigma(z)$. These results illustrate two things. First, with the above definitions of $\mathcal{A}$ and $\mathcal{B}$, there are many more constraints in the IISs than in the RIISs. More importantly, our results highlight the substantial reduction in the number of constraints that are obtained by filtering the set $\sigma(z)$ returned by the IISs.

We illustrate in Table 2 the benefit of starting with an extreme point of $\text{Proj}_z(P)$ rather than P. In both cases, we compute 100 RIISs by filtering from 100 IISs obtained by solving (2) with random costs. To ensure that we get an extreme point of $\text{Proj}_z(P)$, we set to 0 the cost of variables y ($c = 0$) in the former case. In contrast, we sample c randomly to get arbitrary extreme points of P. Table 2 provides for each approach the average size of the RIISs (|RIIS|) and the cardinality of the support of z in the solutions of (2) ($|\sigma(z)|$), as well as the number of iterations (#it) required by Algorithm 1. The last three columns compute the relative reductions in the sizes of RIISs, number of iterations, and support of z when setting $c = 0$. The results illustrate the rather consistent decrease in the cardinality of the support of z when $c = 0$, with 16 out of the 27 instances witnessing a reduction and only two witnessing an increase (of less than 10%). The numbers of iterations follow a similar trend. The size of resulting RIISs are only marginally

name	#RIIS	|RIIS|	#clusters
bgdbg1	12	2.0	12
bgetam	1	1.0	1
bgindy	1	3.0	1
bgprtr	1	5.0	1
box1	1	144.0	1
ceria3d	2	5.5	2
chemcom	1	2.0	1
ex72a	1	73.0	1
ex73a	1	24.0	1
forest6	1	3.0	1
galenet	1	1.0	1
gams30am	1	1.0	1
gams60am	1	1.0	1
gosh	1	1.0	1
greenbea	2	3.0	2
itest2	2	2.0	2
itest6	3	1.3	3
klein1	1	4.0	1
klein2	3	1.0	3
klein3	6	80.0	1
mondou2	3	34.0	3
pang	1	2.0	1
pilot4i	1	1.0	1
qual	1	8.0	1
reactor	2	1.0	2
vol1	1	8.0	1
woodinfe	2	1.0	2

Table 3 RIISs obtained through Algorithm 2.

affected by setting $c = 0$. Overall, Table 2 illustrates that, while working with the projection may be a bit faster than working with the full polyhedron when looking for an RIIS, the projection does not necessarily lead to smaller RIISs.

Table 3 illustrates Algorithm 3 on the same instances as before. The table reports the number of RIISs (#RIIS) found by the algorithm and their average sizes (|RIIS|). In addition, in the last column, we report the number of clusters of RIISs returned by the algorithm.

To better discuss the results reported in the table, we draw attention on another classical issue in infeasibility analysis. It is often of interest to identify minimal subsets of constraints that need to be removed from $\mathcal{S}$ to recover feasibility (Chinneck, 2008a). Those coincide with sets of constraints that cover all IISs. This definition naturally extends to our context, leading to covers of RIISs. Now, observe that the numbers of RIISs (#RIIS) and clusters of RIISs displayed in Table 3 coincide for all but one instance (klein3). Hence, for all these instances, the RIISs returned by Algorithm 3 are disjoint, and thus minimum covers must contain at least one constraint that belongs to each one of these RIISs. Stated otherwise, the result reported in column #RIIS provides a lower bound on the cardinality of the minimum RIIS cover for each instance except klein3.

Comparing column #clusters of Table 3 with column "Minimum cover cardinality" of Table 7.1 from Chinneck (2008a), we see that for three of these instances (itest6, klein2, and reactor) the cardinality of the minimum cover is larger for RIISs than for IISs, because one or more of the constraints in the IISs cover are in set $\mathcal{A}$. Therefore, in the context where the constraints of $\mathcal{A}$ are known to be mandatory

in the description of the problem, the results for these three instances indicate that repairing (for instance, by removing) the constraints included in a minimum IIS cover would not be enough to recover feasibility. In contrast, a minimum RIIS cover is composed of a set of constraints that can be repaired to recover feasibility.

7 Conclusion

In this work, we have formalized the concept of RIIS, discussing how it is related to the classical IIS, and providing a practical algorithm to compute them. We have also focused on the IISs and RIISs that can be obtained from the *Phase I* of the simplex algorithm, answering a question raised by Chinneck and Dravnieks (1991) related to the covering of clusters of IISs.

While our motivation has been driven by detecting infeasibility in linear programs, we believe RIISs can also be useful in integer programming wherein understanding infeasibility is also at the core of several cutting planes algorithms. For instance, Codato and Fischetti (2006); Fischetti et al. (2010) solve a specific Benders' decomposition that keeps all the binary variables in the master problem and search for particular IISs in the subproblem. Another application is in sparse approximation problems, which are typically cast as MILPs that minimize the number of binary variables taking a value different from 0. Enforcing these variables to be equal to 0 leads to a linear system where the latter constraints belong to $\mathcal{B}$. Developing efficient cutting plane algorithms leveraging the cuts derived from the RIISs could be an interesting venue for future work.

References

Edoardo Amaldi, Marc E. Pfetsch, and Leslie E. Trotter. On the maximum feasible subsystem problem, IISs and IIS-hypergraphs. *Mathematical Programming, Series B*, 95(3):533–554, 2003. ISSN 00255610. doi: 10.1007/s10107-002-0363-5.

Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B. Shah. Julia: A Fresh Approach to Numerical Computing. *SIAM Review*, 59(1):65–98, 2017. ISSN 0036-1445, 1095-7200. doi: 10.1137/141000671. URL `https://epubs.siam.org/doi/10.1137/141000671`.

John W. Chinneck. Finding a useful subset of constraints for analysis in an infeasible linear program. *INFORMS Journal on Computing*, 9(2):164–174, 1997. ISSN 10919856.

John W. Chinneck. *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*. Springer, 2008a. ISBN 978-0-387-74931-0. Series Title: International Series in Operations Research & Management Science.

John W. Chinneck. Isolating Infeasibility. In *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*, pages 93–157. Springer, 2008b. Series Title: International Series in Operations Research & Management Science.

John W. Chinneck and Erik W. Dravnieks. Locating Minimal Infeasible Constraint Sets in Linear Programs. *ORSA Journal on Computing*, 3(2):157–168, 1991. ISSN 0899-1499. doi: 10.1287/ijoc.3.2.157. URL `http://pubsonline.informs.org/doi/abs/10.1287/ijoc.3.2.157`.

Gianni Codato and Matteo Fischetti. Combinatorial Benders' Cuts for Mixed-Integer Linear Programming. *Operations Research*, 54(4):756–766, 2006. ISSN 0030-364X. doi: 10.1287/opre.1060.0286.

George B Dantzig and Mukund N Thapa. *Linear Programming , 1 : Introduction*. Springer Series in Operations Research. Springer-Verlag, 1997. ISBN 0-387-94833-3.

Matteo Fischetti, Domenico Salvagnin, and Arrigo Zanette. A note on the selection of Benders' cuts. *Mathematical Programming*, 124(1-2):175–182, 2010. ISSN 00255610. doi: 10.1007/s10107-010-0365-7.

John Gleeson and Jennifer Ryan. Identifying Minimally Infeasible Subsystems of Inequalities. *ORSA Journal on Computing*, 2(1):61–63, 1990. ISSN 0899-1499. doi: 10.1287/ijoc.2.1.61. URL `http://pubsonline.informs.org/doi/abs/10.1287/ijoc.2.1.61`.

Mark Parker and Jennifer Ryan. Finding the minimum weight IIS cover of an infeasible system of linear inequalities. *Annals of Mathematics and Artificial Intelligence*, 17:107–126, 1996.

Marc E Pfetsch. Branch-and-cut for the maximum feasible subsystem problem. *SIAM Journal on Optimization*, 19(1):21–38, 2008.

M. Tamiz, S.F. Mardle, and D.F. Jones. Detecting IIS in Infeasible Linear Programmes using Techniques from Goal Programming. *Computers & Operations Research*, 23(2):113–229, 1996.

J. N M van Loon. Irreducibly inconsistent systems of linear inequalities. *European Journal of Operational Research*, 8(3):283–288, 1981. ISSN 03772217. doi: 10.1016/0377-2217(81)90177-6.