Identifying infeasible subsets of linear inequalities that are irreducible with respect to a given subset of the inequalities

Jérémy Omer, Michael Poss

To cite this version:

Jérémy Omer, Michael Poss. Identifying infeasible subsets of linear inequalities that are irreducible with respect to a given subset of the inequalities. 2019. hal-02290959

HAL Id: hal-02290959
https://hal.archives-ouvertes.fr/hal-02290959
Preprint submitted on 18 Sep 2019

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.
Identifying infeasible subsets of linear inequalities that are irreducible with respect to a given subset of the 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 subsets 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 (8)-IISs as infeasible subsets 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 infeasible subsets irreducible with respect to a subsystem, making the link with the dual polytope commonly used in the detection of IISs. We then turn to the study of the (8)-IISs that can be obtained from the Phase I of the simplex algorithm. We answer an open question regarding the covering of the clusters of such (8)-IISs and deduce a practical algorithm to find these covering (8)-IISs. 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 subsets of infeasible inequalities. The best that can be done in this direction is identifying irreducible infeasible subsets of inequalities (IISs), i.e., infeasible subsets whose proper subsets are all feasible. The isolation of infeasibility has an obvious applica-
tion in the diagnosis of infeasibility for practitioners who would like to understand why their model is infeasible and how this can be easily fixed.

IISs returned by high-dimensional 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 eluded. 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 other hand, bounds and non-negativity constraints, and on the other hand, all other inequalities, called functional constraints therein. He then looks 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 filtering method from Chinneck and Dravnieks (1991).

The purpose of this study is two-fold. First we wish to generalize and formalize the work of Chinneck and Dravnieks (1991) by focusing on the search of infeasible subset of inequalities that are irreducible with respect to only a subsystem of linear inequalities. Stated otherwise, let \((S)\) be a system of inequalities and assume that we wish to focus our analysis on some subsystem of \((S)\), denoted \((B)\): an infeasible system irreducible with respect to \((B)\) is a subsystem of \((S)\) that becomes feasible if any inequality of \((B)\) is removed from it. This allows to achieve minimality with respect to the subsystem of interest. Second, we focus on the \((B)\)-IISs that can be obtained from the optimal solution of the Phase I of the simplex algorithm. We show that these cover all clusters of \((B)\)-IISs, thus answering and generalizing a conjecture of Chinneck and Dravnieks (1991).

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 \((B)\)-IIS is then introduced in Section 3 and the dual 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 set of \(m\) linear inequalities with unknown \(x \in \mathbb{R}^n\):

\[
(S) : Ax \leq a, Bx \leq b,
\]

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 \((A)\) and \((B)\).

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 \( \sigma(x) \).

We will refer to a subsystem by the set that indexes the corresponding inequalities surrounded with brackets. For instance, the subsystem \([I]\) of \((A)\) consists of the inequalities \( A_I x \leq a_I \). By extension, the subsystem \([I|J]\) of \((S)\) is \( \{ A_I x \leq a_I, B_J x \leq b_J \} \).

## 2 Irreducible infeasible subsystems of linear inequalities

We first recall the classical definition of an irreducible infeasible set (van Loon (1981)).

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

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

**Proposition 1** It is possible to extract an IIS from any infeasible subset of linear inequalities.

*Proof* The proof is tightly connected to the filtering algorithm of Chinneck and Dravnieks (1991), which iteratively constructs an IIS by removing constraints from an infeasible set.

Let \( I \subseteq \{1, \ldots, m\} \) index an infeasible subset of inequalities of \((S)\). As in the filtering algorithm, we remove inequalities from \( I \) while we can find one that can be removed without making the new system feasible. Denoting \( J \) the indices of all the inequalities that have been removed, it is straightforward that \( [I \setminus J] \) is an IIS of \((S)\). Indeed, the filtering algorithm conserves the infeasibility of the subsystem and it stops when no inequality can be removed from the subsystem without making it feasible.

One noteworthy consequence of Proposition 1 is that 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 selects inequalities that trigger infeasibility. These two algorithms have been at the origin of a series of computational improvements reviewed by Chinneck (2008). This lead 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 polyhedral characterization of the set of IISs.
Theorem 1 (Gleeson and Ryan (1990)) The indices of the IISs of (S) are in one-to-one correspondence with the supports of the vertices of the polyhedron

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

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

Proof (Sketch of proof) Farkas theorem of the Alternative shows that the support of any point of P describes an infeasible subset of (S). The second step of the proof uses the characterization of an extreme point to show that an infeasible subset of (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 yields a vertex of P.

Fischetti et al. (2010) then 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 Infeasible subsets irreducible with respect to a subsystem

What we intend to accomplish is analyse the infeasibility of (S) with a focus on the role of the inequalities of (B). Specifically, we assume that inequalities of (A) hold and wish to understand how subsets of inequalities of (B) lead to infeasibilities.

Definition 3 Let \([J]\) be a subsystem of (B). We say that \([J]\) is an irreducible infeasible subset of inequalities with respect to (B) if and only if it satisfies

\[ \{1, \ldots, m_A\} \mid J \] is infeasible and \([\{1, \ldots, m_A\} \setminus J']\) is feasible, \(\forall J' \subseteq J\) (1)

In the remainder of the article, we will call such subsystem a (B)-IIS.

The above definition allows to generalize that of an IIS, because an IIS is actually irreducible with respect to the complete system (S). We illustrate in the the following example the relationship between IISs and (B)-IISs.
Example 1 Let \((S)\) be defined by the subsystems
\[
(A) : \begin{cases} 
0 \leq x_1 \leq \frac{1}{2} \\
0 \leq x_2 \leq 1 \\
-x_1 + x_2 \leq \frac{1}{2}
\end{cases}
\quad \text{and} \quad
(B) : \begin{cases} 
3x_1 + x_2 \geq 3 \\
-x_1 + x_2 \geq \frac{3}{2}
\end{cases}
\]
A graphic representation of the inequalities of \((A)\) and \((B)\) appears in Figure 1.

The enumeration of the IISs of the complete system provides the following five subsets 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 \((B)\)-IISs:

- \(\{-x_1 + x_2 \geq \frac{3}{2}\}\), 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 a \((B)\)-IIS in an analysis of infeasibility that focuses on \((B)\) since it contains the whole system. Also, the first two IISs would actually provide redundant information with respect to \((B)\).

Remark 1 By definition, for all \((B)\)-IIS, \([J]\), there exists an IIS of \((S)\), \([I|J]\). Reciprocally, there is no guarantee that \([J]\) is a \((B)\)-IIS if \([I|J]\) is an IIS of \((S)\). This is illustrated in the last IIS from Example 1.
For illustration, we treat several simple cases.

- If \( B = \emptyset \), then there is no \((B)\)-IIS.
- If \( A = \emptyset \), then the search for a \((B)\)-IIS comes down to the search for IISs of \((S)\).
- If \( A \) is infeasible, then there is no \((B)\)-IIS, because \{\( x \mid Ax \leq a, B_Jx \leq b_J \)\} = \( \emptyset \).
- If \( B \) consists of a single inequality, then there is a \((B)\)-IIS if and only if \( A \) is feasible. In this case, \((B)\) is the only \((B)\)-IIS.

To leave the above cases aside in the rest of the presentation, we assume that \((A)\) is feasible and \((B)\) includes at least two inequalities.

4 Dual polyhedron and the filtering method

Let \( P := \{(y,z) \in \mathbb{R}^{m_A} \times \mathbb{R}^{m_B} \mid y^TA + z^TB = 0, y^Ta + z^Tb \leq -1, y \geq 0, z \geq 0\} \), be the dual polyhedron that appears in the characterization of Theorem 1. We have seen that an extreme point of \( P \) can be found by minimizing \( c^Ty + d^Tz \) 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 aims at producing IISs that include a minimum number of inequalities of \((B)\). 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 polytope characterization of IISs can be adapted to narrow the search for \((B)\)-IISs.

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 \( z \),

\[
\text{Proj}_z(P) := \left\{ z \geq 0 \mid \exists y \geq 0, y^TA + z^TB = 0, y^Ta + z^Tb \leq -1 \right\}.
\]

As a projection of \( P \) on a linear subspace, for all extreme point of \( \text{Proj}_z(P) \), \( \hat{z} \), there is \( \hat{y} \) such that \((\hat{y}, \hat{z})\) is an extreme point of \( P \). However, it is not reciprocally true that for all extreme point of \( P \), \((\hat{y}, \hat{z})\), \( \hat{z} \) is an extreme point of \( \text{Proj}_z(P) \). The only general result is that \( \hat{z} \) is in the convex envelope of the extreme points of \( \text{Proj}_z(P) \). One can consider the projection of a pyramid on the plane of its base for a counterexample.

4.1 Link between the vertices of \( \text{Proj}_z(P) \) and \((B)\)-IISs

We investigate next whether Theorem 4 transposes to \((B)\)-IISs. The following result shows that any \((B)\)-IIS corresponds to an extreme point of \( \text{Proj}_z(P) \).

**Proposition 2** Let \([J]\) be a \((B)\)-IIS, then there is an extreme point of \( \text{Proj}_z(P) \) whose support coincides with \( J \).
Example 3

We focus once again on the system defined in Example 1. Let \( (\mathcal{A}) \): 

\[
\begin{align*}
-x_1 & \leq 0 & [y_1] \\
x_1 & \leq \frac{1}{2} & [y_2] \\
-x_2 & \leq 0 & [y_3] \\
x_2 & \leq 1 & [y_4] \\
-x_1 + x_2 & \leq \frac{1}{2} & [y_5]
\end{align*}
\]

and \( (\mathcal{B}) \):

\[
\begin{align*}
-3x_1 - x_2 & \leq 3 & [z_1] \\
x_1 - x_2 & \leq -\frac{3}{2} & [z_2]
\end{align*}
\]

Proof

If \([J]\) is a \((\mathcal{B})\)-IIS, there is \( I \subseteq \{1, \ldots, m_A\} \) such that \([I,J]\) is an IIS of \((\mathcal{S})\). By Theorem \([\star]\), there is an extreme point of \( P \), \((\hat{y}, \hat{z})\), whose support is given by \( I \cup J \). Now, assume that \( \hat{z} \) is not an extreme point of \( \text{Proj}_z(P) \). In such case, we can write \( \hat{z} \) as a convex combination of \( K \leq m_B + 1 \) extreme points of \( \text{Proj}_z(P) \):

\[
\hat{z} = \sum_{k=1}^{K} \alpha_k \bar{z}^k, \quad \sum_{k=1}^{K} \alpha_k = 1, \quad 0 \leq \alpha_k \leq 1, \quad k = 1, \ldots, K.
\]

Let \( k \in \{1, \ldots, K\} \). Since \( z \geq 0 \) for all \( z \in \text{Proj}_z(P) \), the support of \( \bar{z}^k \) is necessarily included in that of \( \hat{z} \), i.e., \( \sigma(\bar{z}^k) \subseteq J \). Besides, there is \( y^k \) such that \((y^k, \bar{z}^k)\) is an extreme point of \( P \), so \( \sigma(\sigma(y^k))\sigma(\bar{z}^k) \) is an IIS of \((\mathcal{S})\). Given that \([J]\) is a \((\mathcal{B})\)-IIS of \((\mathcal{S})\), we get that \( J \subseteq \sigma(\bar{z}^k) \), hence \( \sigma(\bar{z}^k) = J \).

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

Proposition \([\star]\) guarantees that we can focus on the vertices of \( \text{Proj}_z(P) \) instead of those of \( P \), since no \((\mathcal{B})\)-IIS will be left aside by doing so. Unfortunately, the reciprocal of the proposition does not hold as there may exist a vertex of \( \text{Proj}_z(P) \) that is not supported by a \((\mathcal{B})\)-IIS. This is illustrated in the example below.

Example 2

Let \((\mathcal{S})\) be defined by the subsystems

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

The dual 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, \((0, \frac{4}{5}, \frac{2}{5}, \frac{2}{5})\) and \((1, 1, 0, 1)\) whose projections on the space of \( z \) variables are \((\frac{2}{5}, \frac{2}{5})\) and \((0, 1)\). It so happens that \((\frac{2}{5}, \frac{2}{5})\) and \((0, 1)\) are also the only vertices of \( \text{Proj}_z(P) \). However, there is only one \((\mathcal{B})\)-IIS that consists of the last constraint of \((\mathcal{B})\): \{-\(x_1 - x_2 \leq -3\}\}.

There are also cases where all vertices of \( \text{Proj}_z(P) \) lead to \((\mathcal{B})\)-IISs, as shown in the example below. The following example also illustrates that there are cases where \([I,J]\) is an IIS of \((\mathcal{S})\), but no vertex of \( \text{Proj}_z(P) \) is supported by \( J \). This shows that there can be a benefit in focussing on the vertices of \( \text{Proj}_z(P) \) instead of those of \( P \).

Example 3

We focus once again on the system defined in Example \([\star]\).
The dual polyhedron is given by

\[
P = \left\{ (y, z) \in \mathbb{R}^5 \times \mathbb{R}^2 : \begin{array}{l}
- y_1 + y_2 - y_3 - 3z_3 - z_2 = 0, \\
- y_3 + y_4 + 5z_1 - z_2 = 0, \\
\frac{1}{2}y_2 + y_4 + \frac{1}{2}y_5 - 3z_2 - \frac{3}{2}z_2 \leq -1, \\
y \geq 0, z \geq 0
\end{array} \right\}
\]

Using the Polyhedra\(^1\) and CCDLib\(^2\) packages of the Julia language (Bezanson et al. (2017)), we enumerate the vertices of \(P\) as \((0, 0, 0, 0, 1, 0, 1), (2, 0, 0, 2, 0, 0, 2), (0, 3, 0, 2, 0, 2, 0), (0, 0, 0, 2, 2, 0)\) and \((0, 0, 0, \frac{2}{3}, 0, \frac{2}{3}, \frac{2}{3})\), whose supports correspond exactly to the IISs enumerated in Example 1. The projection of these vertices on \(z\) variables yield three subsystems of \((B)\): \((-3z_1 - x_2 \leq -3), \{x_1 - x_2 \leq -\frac{3}{2}\}\) and \((-3z_1 - x_2 \leq -3, x_1 - x_2 \leq -\frac{3}{2})\). In contrast, \(\text{Proj}_z(P)\), has only two vertices, \((0, 1)\) and \((1, 0)\) whose supports correspond exactly to the \((B)\)-IISs of the system.

4.2 Generalization of the filtering and additive methods

As discussed in the previous section, it may be wise to look for a \((B)\)-IIS starting from a vertex of \(\text{Proj}_z(P)\) rather than \(P\). 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

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

The only specificity is that \(c\) should be set to zero and \(d\) needs to be generated randomly.

**Proposition 3** Let \(c \in \mathbb{R}^m_n\) be a vector of mutually independent continuous random variables. Then, if \((\hat{y}, \hat{z}) \in \arg\min\{c^T z : (y, z) \in P\}\), \(\hat{z}\) is a vertex of \(\text{Proj}_z(P)\) with probability 1.

**Proof** Let \(c \in \mathbb{R}^m_n\). Given that \(P \neq \emptyset\) and \(z \geq 0\) for all \((y, z) \in P\), then the linear program \(\min\{c^T z : (y, z) \in P\}\) is feasible and bounded. As a consequence, \(\arg\min\{c^T z : (y, z) \in P\}\) is a nonempty face of \(P\) whose projection on the space of \(z\) is denoted as \(Z_c\). More formally,

\[
Z_c = \text{Proj}_z \left( \arg\min\{c^T z : (y, z) \in P\} \right) \neq \emptyset.
\]

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

\[
\mathbb{P}\left(\dim(Z_c) \geq 1\right) = 0.
\]
Now, assume that \( Z \) is reduced to one point \( \hat{z} \). If \( \hat{z} \) is not a vertex of \( \text{Proj}_z(P) \), there exist two distinct points \( z^1 \) and \( z^2 \) of \( \text{Proj}_z(P) \) such that \( \hat{z} = \alpha z^1 + (1 - \alpha)z^2 \) for some \( \alpha \in [0,1] \). This is only possible if \( c^T z^1 = c^T z^2 = \min \{c^T z : (y,z) \in P\} \), a contradiction. We deduce that with probability 1, \( \arg\min \{c^T z : (y,z) \in P\} \) is reduced to one vertex of \( \text{Proj}_z(P) \).

\( \square \)

Remark 2 Tolerances in optimality and feasibility will necessarily yield a nonzero probability that the solution of \( \min \{c^T z : (y,z) \in P\} \) be a vertex of \( \text{Proj}_z(P) \). In practice though, we will still obtain an IIS from which a \((B)\)-IIS can be found by filtering.

Once a vertex has been found, a \((B)\)-IIS can be found using, for instance, the filtering algorithm described in Algorithm 1. Its validity is guaranteed by the following result.

**Proposition 4** Let \( [J] \) be a subsystem of \((B)\) such that \( \{x \mid Ax \leq a, B_J x \leq b_J\} = \emptyset \), then it is possible to extract at least one \((B)\)-IIS, \( J' \subset J \).

**Proof** We show the result by induction on the size of \( [J] \). If \( [J] \) consists of only one inequality, then it is a \((B)\)-IIS. Now, assume that \( [J] \) includes more than one constraint. If there exists \( J' \subset J \) such that \( \{x \mid Ax \leq a, B_J x \leq b_J\} = \emptyset \), then apply the induction hypothesis on \( [J'] \). Otherwise, \( [J] \) is a \((B)\)-IIS (by definition).

\( \square \)

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 \((B)\)-IIS from a subsystem of \((B)\).

The next section turns to the analysis of the IISs and \((B)\)-IISs that can be deduced from the solution of the Phase I LP.
5 Phase I sensibility 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 so that a trivial feasible solution appears, and minimize the values of the 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 \( A \) : x \geq 0 and \( B \) : 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 where \( A \) : Ax \geq a for some feasible system of linear inequalities Ax \geq a, the Phase I LP considered by Chinneck and Dravnieks (1991) is

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

(3)

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

\[
\text{LD}_I: \begin{cases}
\max & -a^T y + b^T z \\
\text{subject to} & A^T y + B^T z = 0, \\
& y \geq 0, 0 \leq z \leq 1
\end{cases}
\]

(4)

Given that \( A \) is feasible, (3) and (4) both have optimal solutions. In the remainder of this section, \( (x^*, s^*) \) is an extreme optimal solution of LP\(_I\) and \( (y^*, z^*) \) is a complementary extreme optimal solution of 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 a \( (B) \)-IIS cover, i.e., for all \( (B) \)-IIS \( J \), \( \sigma(s^*) \cap J \neq \emptyset \);
2. \( s_j^* > 0 \) only if \( B_j x \leq b_j \) belongs to an IIS;
3. \( y_i^* > 0 \) only if \( A_i x \leq 0 \) belongs to an IIS;
4. if \( J = \sigma(z^*) \), then \( J \) contains a \( (B) \)-IIS.

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. Chinneck and Dravnieks (1991) conjecture that the support of \( (y^*, z^*) \) contains at least one IIS from each cluster. In what follows, we first generalize the definition of cluster to \( (B) \)-IISs, and we show more generally that the support of \( z^* \) contains the indices of at least one \( (B) \)-IIS per cluster to \( (B) \)-IISs.
Definition 4 A set \( C \) of \((B)\)-IISs of \((S)\) is a cluster of \((B)\)-IISs if and only if:

- \( C \neq \emptyset \); 
- For all \((B)\)-IIS \([J]\), if there is \([J'] \in C\) such that \( J \cap J' \neq \emptyset \), then \([J'] \in C\).

Theorem 2 Let \( l \in \{1, \ldots, m_B\} \) be such that \( s_l^* > 0 \). There exists at least one \((B)\)-IIS of \((S)\), \([J]\), such that \( l \in J \) and \( z_l^* > 0 \), \( \forall j \in J \).

Proof By complementarity of the primal and dual solutions, we know that for all \( l \in \sigma(s^*)\), \( l \in \sigma(z^*)\). Let \( \text{LP}_1^*\) be the LP obtained from \( \text{LP}_1\) by keeping only the constraints of \((B)\) indexed by \( \sigma(z^*)\). Denoting \( \bar{s} = s_{\sigma(z^*)} \) and \( \bar{z} = z_{\sigma(z^*)} \), one can readily verify that \((z^*, \bar{s})\) is an optimal solution of \( \text{LP}_1\), by complementarity with the dual solution \((y^*, \bar{z})\).

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 \( \text{LP}_1^*\) guarantees that \( B_l x \leq b \) belongs to an IIS, \([l|J]\), of the constraints of \( \text{LP}_1\). By definition of \( \text{LP}_1^*\), we necessarily have \( J \subseteq \sigma(z^*)\). Hence, we can extract a \((B)\)-IIS \([J']\) from \([l|J]\) such that \( J' \subseteq \sigma(z^*)\). \( \square \)

Corollary 1 The support \( z^*, \sigma(z^*)\), contains the indices of at least one \((B)\)-IIS from each cluster of \((B)\)-IISs.

Proof Let \( C \) be a cluster of \((B)\)-IISs and \( J \in C \). 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 Theorem 2 there is a \((B)\)-IIS, \([J_l]\), such that \( J_l \subseteq \sigma(z^*) \) and \( l \in J_l \). Since \( l \) indexes an inequality involved in at least one \((B)\)-IIS of \( C\), \([J_l]\) must also belong to \( C\), which concludes the proof. \( \square \)

The motivation for considering clusters of \((B)\)-IISs is that they can be more numerous than clusters of IISs. Indeed, two \((B)\)-IISs \( J_1 \) and \( J_2 \) can belong to different clusters of \((B)\)-IISs even though every pair of IISs, \([J_1|J_2]\) and \([J_2|J_2]\), belong to the same cluster of IISs.

In the following result, we specify how the optimal dual solution can be written from a combination of extreme points of \( \tilde{P} \). The result and its constructive proof yield an algorithm that can produce several \((B)\)-IISs (at least one per cluster) 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

\[
 z^* = \sum_{k=1}^{K} \alpha_k z^*_k + \bar{z},
\]

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

- \( \alpha_k > 0 \),
- \( \sigma(z^*_k) \) is a \((B)\)-IIS,
- there is \( j \in \sigma(z^*_k) \) such that \( s_j^* > 0 \).

Moreover, \([\{1, \ldots, m_A\}|\sigma(\bar{z})]\) is a feasible subsystem of \((S)\).
Proof Let \( \bar{z} := z^* \). From the assumption that \((S)\) is not feasible, we know that
\[
a^Ty^* + b^Tz^* = a^Ty^* + b^T\bar{z} < 0.\]
As a consequence, \( \sigma(y^*) \cup \sigma(\bar{z}) \) indexes an infeasible subsystem of \((S)\), and so does \( \{1, \ldots, m_A\} \cup \sigma(\bar{z}) \). This implies that there is an IIS of \((S)\), \([I,J]\), such that \( J \subseteq \sigma(\bar{z}) \) and \([J]\) is a \((B)\)-IIS.

Let \((y^*, z^*)\) be an extreme point of \(P\) such that \( \sigma(z^*) = J \). First, \((x^*, s^*)\) is an optimal solution of \(LP_1\) so \( Ax^* \leq a \) and \( Bx^* - s^* \leq b \). In particular, we have
\[
A_{\sigma(y^*)}x^* \leq a_{\sigma(y^*)} \text{ and } B_{\sigma(x^*)}x^* - s^*_{\sigma(z^*)} \leq b_{\sigma(z^*)}. \]
But, \( [\sigma(y^*) \sigma(z^*)] \) is an infeasible subsystem of \((S)\), so there must be \( j \in \sigma(z^*) \) such that \( s^*_j > 0 \). Second, we have
\[
\sigma(z^*) \subseteq \sigma(\bar{z}) \text{ so there is } \alpha_1 > 0 \text{ such that }
\]
\[
\sigma(\bar{z} - \alpha_1 z^1) \subseteq \sigma(\bar{z}),
\]
\[
\bar{z} - \alpha_1 z^1 \geq 0
\]

Updating \( \bar{z} \) as \( \bar{z} := \bar{z} - \alpha_1 z^1 \), we can apply the above process recursively until \( \{1, \ldots, m_A\} \cup \sigma(\bar{z}) \) is feasible, which 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 IISs as described in Theorem 2. For this, we can follow the algorithm below.

<table>
<thead>
<tr>
<th>Input: The system ((S) : Ax \leq a, Bx \leq b, c \in \mathbb{R}^{m_A}, d \in \mathbb{R}^{m_B};)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ((x^<em>, s^</em>) \leftarrow \text{ an optimal extreme solution of } LP_1;)</td>
</tr>
<tr>
<td>2 ((\bar{y}, \bar{z}) \leftarrow \text{ a complementary dual solution;})</td>
</tr>
<tr>
<td>3 (Z \leftarrow \sigma(\bar{z}); \quad \mathcal{I}_{(B)\text{-IIS}} \leftarrow \emptyset, \quad k \leftarrow 0;)</td>
</tr>
<tr>
<td>4 \textbf{while} ( {1, \ldots, m_A} \cup \sigma(\bar{z}) \text{ is infeasible do})</td>
</tr>
<tr>
<td>5 \quad k \leftarrow k + 1;</td>
</tr>
<tr>
<td>6 \quad \text{// get one one ((B))-IIS from } Z</td>
</tr>
<tr>
<td>7 \quad (I[J]) \leftarrow \text{ an IIS such that } J \subseteq Z;</td>
</tr>
<tr>
<td>8 \quad J \leftarrow \text{ extract a } (B)\text{-IIS from } [I</td>
</tr>
<tr>
<td>9 \quad \mathcal{I}<em>{(B)\text{-IIS}} \leftarrow \mathcal{I}</em>{(B)\text{-IIS}} \cup {J};</td>
</tr>
<tr>
<td>10 \quad \text{// prepare for next iteration}</td>
</tr>
<tr>
<td>11 \quad \text{min}_{I,J} (c^T y + d^T J z_J : A^T y + B^T J z_J = 0, \quad a^T y + b^T J z_J = -1, \quad y \geq 0, \quad z_J \geq 0);</td>
</tr>
<tr>
<td>12 \quad \alpha_k := \text{min}_{I,J} \left{ \frac{c_I}{a^T J} \right};</td>
</tr>
<tr>
<td>13 \quad \bar{z} \leftarrow \bar{z} - \alpha_k z^k;</td>
</tr>
<tr>
<td>14 \quad Z \leftarrow \sigma(\bar{z});</td>
</tr>
<tr>
<td>15 \textbf{return } \mathcal{I}_{(B)\text{-IIS}};</td>
</tr>
</tbody>
</table>

Algorithm 2: Identification of \((B)\)-IISs using Phase I sensitivity analysis

**Corollary 2** At the end of Algorithm 2 \( I_{(B)\text{-IIS}} \) includes at least one \((B)\)-IIS per cluster of \((B)\)-IISs.

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
a \( (B)\)-IIS whose support is included in \( \sigma(\bar{z}) \), and an LP is solved to get a dual solution \( (y^k, z^k) \) such that \( \sigma(z^k) \subseteq \sigma(\bar{z}) \). 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 \( \sigma(z^k) \) is a \( (B)\)-IIS, and \( \sigma(\bar{z}) \) is a feasible subsystem of \( (S) \).

Now, let \( C \) be a cluster of \( (B)\)-IISs. By Corollary 1, there is \( J \in C \) such that \( J \subseteq \sigma(z^*) \). Given that \( \sigma(z^*) \) 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 a \( (B)\)-IIS, this means that \( \sigma(z^k) \in C \). \( \square \)

6 Computational experiments

We detail below the results of our experiments realized on infeasible instances made available by Csaba Mészáros\(^3\) mentioned in Chinneck (2008), among others. As these instances consider classical IISs, instead of \( (B)\)-IISs, we needed to create a decomposition of the linear systems involved into systems \( (A) \) and \( (B) \). For each instance, we decided to put in \( (A) \) all bounds on the variables, as well as the equalities with right-hand-side equal to 0, suggesting that the latter constraints model the physics of the systems, which must be satisfied. All other inequalities and equalities form the system \( (B) \). The resulting instances are described in Table 1.

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

The main motivation behind the introduction of \( (B)\)-IISs 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 focusing on the projection, e.g., \( c \) may be different from 0). We then extract one \( (B)\)-IIS from each IIS with the filtering algorithm described in Algorithm 1. Table 1 reports the average numbers of constraints returned in IISs (avgIIS), constraints of the IIS that belong to \( (B) \) (\( |\sigma(z)| \)), and constraints in \( (B)\)-IISs (avgBIIS). 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 \( (A) \) and \( (B) \), there are many more constraints in the IISs than in the \( (B)\)-IISs. 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.

\(^3\) Every instance can be downloaded at http://old.sztaki.hu/~meszaros/public_ftp/lptestset/infeas/
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 \((B)\)-IISs 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 the average size of the \((B)\)-IISs (avgBIIS) returned by each approach, as well as the number of iterations required by Algorithm 1 (iterations) and the cardinality of the support of \( z \) in the solutions of (2) (\( |\sigma(z)| \)). The last three columns compute the relative reductions in the sizes of \((B)\)-IISs, 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 13 out of the 27 instances witnessing a reduction. The numbers of iterations follow a similar trend. The size of resulting \((B)\)-IISs are only marginally affected by setting \( c = 0 \). Overall, Table 2 confirms that working with the projection is a bit more efficient than working with the full polytope when looking for a \((B)\)-IIS.

Table 3 illustrates Algorithm 2 on the same instances as before. The table reports the number of \((B)\)-IISs (nBIISs) found by the algorithm and their average sizes. In addition, by application of Corollary 1 we are able to compute the number of clusters of \((B)\)-IISs in each instance. We report it in the last column of the table. The results show that for 10 out of 27 instances, more than one \((B)\)-IIS is found by this algorithm, and up to 22 of them for instance \textit{bgdbg1}. It is also interesting to observe that most of these instances actually exhibit a small number of clusters.

### Table 1

| name            | nvar | nctr | nctrA | nctrB | avgIIS | \(|\sigma(z)|\) | avgBIIS | reduction (%) |
|-----------------|------|------|-------|-------|--------|----------------|---------|---------------|
|\textit{bgdbg1.mps} | 407  | 348  | 126   | 222   | 4.1    | 3.3           | 3.1     | 6.3           |
|\textit{bggetam.mps} | 688  | 400  | 272   | 128   | 20.7   | 4             | 1       | 75.2          |
|\textit{bgindy.mps}  | 10116| 2671 | 1907  | 764   | 152    | 2             | 2       | 0             |
|\textit{box1.mps}    | 34    | 20   | 14    | 6     | 12.3   | 2.1           | 2       | 1.5           |
|\textit{ceria3d.mps} | 824  | 3576 | 0     | 3576  | 149.9  | 149.9         | 149.9   | 0             |
|\textit{chemcom.mps} | 720  | 288  | 250   | 38    | 36.5   | 4             | 2       | 49.7          |
|\textit{ex72a.mps}   | 215   | 197  | 0     | 197   | 59.2   | 58.2          | 58.2    | 0.1           |
|\textit{ex73a.mps}   | 211   | 193  | 0     | 193   | 25.3   | 24.3          | 24      | 1             |
|\textit{forest6.mps} | 95    | 66   | 30    | 36    | 82.8   | 36            | 27      | 24.9          |
|\textit{gamenet.mps} | 8     | 8    | 2     | 6     | 5      | 2            | 1       | 50.2          |
|\textit{gams30am.mps}| 361   | 714  | 1     | 713   | 21.8   | 21.8          | 1       | 95.3          |
|\textit{gams30bm.mps}| 361   | 714  | 1     | 713   | 21.8   | 21.8          | 1       | 95.3          |
|\textit{gosh.mps}    | 10733 | 3792 | 1070  | 2722  | 9      | 6            | 6       | 0             |
|\textit{greenbea.mps}| 5405  | 2393 | 0     | 2393  | 51.8   | 6.7           | 1       | 83.8          |
|\textit{itest2.mps}  | 4     | 9    | 0     | 9     | 3      | 3            | 3       | 0             |
|\textit{itest6.mps}  | 8     | 11   | 2     | 9     | 3.7    | 3.4           | 2       | 37.5          |
|\textit{klein1.mps}  | 54    | 54   | 0     | 54    | 51     | 51            | 50      | 2             |
|\textit{klein2.mps}  | 54    | 477  | 0     | 477   | 53.8   | 53.8          | 49.2    | 8.6           |
|\textit{klein3.mps}  | 88    | 994  | 0     | 994   | 87.1   | 87.1          | 78.6    | 9.8           |
|\textit{mondou2.mps} | 604   | 312  | 117   | 195   | 47.8   | 24.4          | 21      | 13            |
|\textit{pang.mps}    | 460   | 361  | 80    | 281   | 18.6   | 9.6           | 6       | 36.4          |
|\textit{pilot4i.mps} | 1000  | 410  | 210   | 200   | 223.8  | 30.4          | 1       | 90.1          |
|\textit{qual.mps}    | 464   | 323  | 217   | 106   | 242.7  | 17            | 8       | 52.9          |
|\textit{reactor.mps} | 637   | 318  | 0     | 318   | 8.2    | 1            | 1       | 0             |
|\textit{vol1.mps}    | 464   | 323  | 217   | 106   | 236.6  | 14.5          | 8       | 44.4          |
|\textit{woodine.mps} | 89    | 35   | 0     | 35    | 2      | 1            | 1       | 0             |

Table 1  Instance characteristics, averages sizes of IISs and (\(B\))-IISs.
Table 2 Obtaining \((\mathcal{B})\)-IISs from extreme points of \(P\) or \(\text{Proj}_z(P)\).

<table>
<thead>
<tr>
<th>name</th>
<th>(\text{avgBIIS}_P )</th>
<th>(\text{proj}_z(P)) iterations</th>
<th>(\text{avgBIIS}_{\text{Proj}_z(P)} )</th>
<th>(\text{proj}_z(\text{Proj}_z(P))) iterations</th>
<th>(\sigma(\text{proj}_z(P)))</th>
<th>(\sigma(\text{proj}_z(\text{Proj}_z(P))))</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgdbg1</td>
<td>3.1</td>
<td>2.4</td>
<td>3.3</td>
<td>3.3</td>
<td>22.6</td>
<td>9.1</td>
</tr>
<tr>
<td>bggetam</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>bgindly</td>
<td>2</td>
<td>2</td>
<td>2.4</td>
<td>2</td>
<td>2.5</td>
<td>0</td>
</tr>
<tr>
<td>bgprtr</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>box1</td>
<td>8</td>
<td>8</td>
<td>8.2</td>
<td>8</td>
<td>8.3</td>
<td>8</td>
</tr>
<tr>
<td>cereal3d</td>
<td>149.9</td>
<td>149.9</td>
<td>149.9</td>
<td>149.9</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>chemcom</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>3.9</td>
<td>4</td>
<td>3.9</td>
</tr>
<tr>
<td>ex72a</td>
<td>58.2</td>
<td>58.2</td>
<td>58.2</td>
<td>58.2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ex73a</td>
<td>24</td>
<td>24</td>
<td>24.2</td>
<td>24</td>
<td>24</td>
<td>0</td>
</tr>
<tr>
<td>forest6</td>
<td>27</td>
<td>27</td>
<td>33</td>
<td>32</td>
<td>36</td>
<td>34.5</td>
</tr>
<tr>
<td>galenet</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>gams30am</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>10.5</td>
<td>10.5</td>
</tr>
<tr>
<td>gams60am</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>21.8</td>
<td>21.8</td>
</tr>
<tr>
<td>gosh</td>
<td>4</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>itest2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>itest6</td>
<td>2</td>
<td>2</td>
<td>2.8</td>
<td>2</td>
<td>3.4</td>
<td>2</td>
</tr>
<tr>
<td>klein1</td>
<td>50</td>
<td>50</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
</tr>
<tr>
<td>klein2</td>
<td>49.2</td>
<td>49.2</td>
<td>52.2</td>
<td>52.2</td>
<td>53.8</td>
<td>53.8</td>
</tr>
<tr>
<td>klein3</td>
<td>78.6</td>
<td>78.6</td>
<td>82.2</td>
<td>82.2</td>
<td>87.1</td>
<td>87.1</td>
</tr>
<tr>
<td>mondo2</td>
<td>21</td>
<td>20.3</td>
<td>23.6</td>
<td>22.6</td>
<td>24.4</td>
<td>22.8</td>
</tr>
<tr>
<td>pilot4i</td>
<td>1</td>
<td>1</td>
<td>2.4</td>
<td>1.3</td>
<td>30.4</td>
<td>1.5</td>
</tr>
<tr>
<td>qual</td>
<td>8</td>
<td>8</td>
<td>14</td>
<td>13.2</td>
<td>17</td>
<td>15.8</td>
</tr>
<tr>
<td>reactor</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>vol1</td>
<td>8</td>
<td>8</td>
<td>13.4</td>
<td>13.2</td>
<td>14.5</td>
<td>14.5</td>
</tr>
<tr>
<td>woodinfe</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>average</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>6.7</td>
</tr>
</tbody>
</table>

of \((\mathcal{B})\)-IISs. Only six instances have more than one cluster and only \(bgdbg1\) has more than three. This explains why Algorithm 2 generates only a small number of \((\mathcal{B})\)-IISs for most instances.

7 Conclusion

In this work, we have formalized the concept of \((\mathcal{B})\)-IIS, 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 \((\mathcal{B})\)-IISs that can be obtained from the \textit{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 \((\mathcal{B})\)-IISs 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 (B). Developing efficient cutting plane algorithms leveraging the cuts derived from the (B)-IISs could be an interesting venue for future work.

<table>
<thead>
<tr>
<th>name</th>
<th>nBISs</th>
<th>avgBISs</th>
<th>nb clusters</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgdbg1</td>
<td>22</td>
<td>2.5</td>
<td>10</td>
</tr>
<tr>
<td>bggetam</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>bgindy</td>
<td>14</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>bgprtr</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>box1</td>
<td>1</td>
<td>144</td>
<td>1</td>
</tr>
<tr>
<td>ceria3d</td>
<td>1</td>
<td>157</td>
<td>1</td>
</tr>
<tr>
<td>chemecon</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>ex72a</td>
<td>1</td>
<td>73</td>
<td>1</td>
</tr>
<tr>
<td>ex73a</td>
<td>1</td>
<td>24</td>
<td>1</td>
</tr>
<tr>
<td>forest6</td>
<td>4</td>
<td>26</td>
<td>1</td>
</tr>
<tr>
<td>galenet</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>gams50am</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>gams80am</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>gosh</td>
<td>1</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>greenbea</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>itest2</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>itest6</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>klein1</td>
<td>1</td>
<td>50</td>
<td>1</td>
</tr>
<tr>
<td>klein2</td>
<td>1</td>
<td>50</td>
<td>1</td>
</tr>
<tr>
<td>klein3</td>
<td>2</td>
<td>80.5</td>
<td>1</td>
</tr>
<tr>
<td>mondou2</td>
<td>3</td>
<td>37</td>
<td>3</td>
</tr>
<tr>
<td>pang</td>
<td>1</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>pilot4i</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>qual</td>
<td>1</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>reactor</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>vol1</td>
<td>1</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>woodine</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Table 3  (B)-IISs obtained through Algorithm 2.

References


