# Equality Set Projection:
# A new algorithm for the projection of polytopes in halfspace representation

Colin N. Jones, Eric C. Kerrigan and
Jan M. Maciejowski

CUED/F-INFENG/TR.463

March 15, 2004

# Equality Set Projection: A new algorithm for the projection of polytopes in halfspace representation

Colin N. Jones, Eric C. Kerrigan and Jan M. Maciejowski
Department of Engineering
University of Cambridge
Cambridge CB2 1PZ
United Kingdom
{cnj22, eck21, jmm}@eng.cam.ac.uk

Technical Report CUED/F-INFENG/TR.463
March 15, 2004

**Abstract**

In this paper we introduce a new algorithm called Equality Set Projection (ESP) for computing the orthogonal projection of bounded, convex polytopes. Our solution addresses the case where the input polytope is represented as the intersection of a finite number of halfplanes and its projection is given in an irredundant halfspace form. Unlike many existing approaches, the key advantage offered by ESP is its output sensitivity, i.e., its complexity is a function of the number of facets in the projection of the polytope. This feature makes it particularly suited for many problems of theoretical and practical importance in which the number of vertices far exceeds the number of facets. Further, it is shown that for non-degenerate polytopes of fixed size (dimension and number of facets) the complexity is linear in the number of facets in the projection. Numerical results are presented that demonstrate that high dimensional polytopes can be projected efficiently.

## 1 Introduction

Polytopes provide a useful representation for the linear constraints that appear in diverse fields such as control, finance and optimisation. The calculation of the orthogonal projection of a polytope is a fundamental operation that arises in many applications. For example, in control theory, projection is required for reachability analysis [6] and in decision theory for the elimination of existential quantifiers [17]. It can be shown that the calculation of affine maps or Minkowski sums of polytopes are both equivalent to orthogonal projection [11], making a projection algorithm a necessary tool for working with polytopes.

It is well known that polytopes can be represented in two forms: as the convex combination of a finite number of vertices and as the intersection of a finite number of halfspaces. For any $d$-dimensional polytope, represented as the intersection of $q$ halfspaces, the number of vertices required to describe the same polytope is $\mathcal{O}(q^{\lfloor \frac{d}{2} \rfloor})$ in the worst case. Polytopes that exhibit an exponential relationship are common in various fields. For example, a hypercube in $d$-dimensions can be described by $2d$ halfspaces or $2^d$ vertices.

Vertex representation is appropriate in some applications, while in others halfspace is preferred. In this paper, we are interested in polytopes that are in halfspace representation and whose projection should also be given in halfspace form. As the vertex representation of the given polytope can be exponentially more complex than the halfspace representation, we ideally want the complexity of the projection algorithm to depend only on the number of halfspaces, and never to compute the vertices. As the effects of this exponential behaviour do not become significant until the dimensionality of the problem is large, this algorithm is most suited to the projection of high dimensional polytopes.

In this paper we give a new algorithm, dubbed Equality Set Projection (ESP), for the computation of the projection of a bounded polytope described as the intersection of a finite number of halfspaces in arbitrary dimension. We do not make the assumption that the polytope is in general position or that the description is irredundant, although the description of the projection returned by the algorithm is irredundant. It will be shown that for a polytope of fixed size (dimension and number of halfspaces), ESP computes the projection using a number of linear programs that is linear in the number of halfspaces in the projection.

## 1.1 Related Work

We briefly review the literature on projection methods and examine its relation to our work. Current projection methods that can operate in general dimensions can be grouped into three classes: Fourier elimination, block elimination and vertex based approaches. While each approach can be effective for a particular class of polytopes, there is as yet no algorithm whose complexity is a function of the number of halfspaces required to describe the projection. Here, we give a brief overview of these approaches and their derivatives.

Fourier-Motzkin elimination was originally described by Fourier in 1824 and has been improved many times since. This approach can be thought of as the analogue of Gaussian elimination for linear inequalities. At each iteration of the algorithm the polytope is recursively projected by one dimension until the desired dimension is reached. The primary limitation of Fourier-Motzkin elimination is that it generates many redundant constraints at each iteration. It is not practical to remove the redundancies at each step, although modifications of the algorithm due to Černikov [16] in 1963 greatly improved the efficiency by identifying many redundant constraints with very little added work. For some polytopes, algorithms based on Fourier elimination can be efficient and recent work [10] has improved the average case complexity for very sparse constraints. However, for a polytope described by $q$ halfspaces that is to be projected down by $k$ dimensions, the time complexity of Fourier elimination is $\mathcal{O}(q^{2^k})$, often making it unusable even for small problems.

In [13], a modification of Fourier's method is proposed in which a set of $k+1$ constraints is selected and all $k$ dimensions of this set that are to be projected are removed using Gaussian elimination. There are, however, $\binom{q}{k+1}$ sets of constraints that must be considered making this algorithm only suitable for very specific applications.

The second well known approach is block elimination. In this method a polyhedron is defined called the projection cone. The extreme rays of this cone can then be used to find the defining halfspaces of the projection. Methods available for computing the extreme rays, such as the double description algorithm [8] or the reverse search approach [2], are exponential in the worst case and as with Fourier elimination, this approach may generate a large number of redundant inequalities that need to be removed.

Recent work by Balas [3] has shown that if certain invertibility conditions are satisfied, then every extreme ray of the projection cone generates an irredundant constraint of the projection. This observation has been extended to polytopes that do not satisfy these conditions by the introduction of a transformation of the cone such that each ray of the projection of the transformed cone corresponds to a constraint of the projection of the polytope. The limitation of this approach is that the calculation of the extreme rays of the projection of the transformed cone may be very difficult, although it offers an important insight into the structure of projection.

The final class of approaches covers a variety of methods that compute vertices of the projection. While these approaches are suitable for a certain class of polytopes, we are interested here in polytopes in which the vertices greatly outnumber the inequalities. As a result, any approach that requires the enumeration of the vertices of either the polytope or its projection can be as much as exponentially slower for this class than an approach that considers only inequalities.

All vertices of the polytope can be computed using a vertex enumeration algorithm, each vertex can then be trivially projected, before a convex hull algorithm is used to calculate the inequality constraints of the projection. Vertex enumeration and convex hulls can be computed using the same algorithms (for example, [2, 8]). This approach can be efficient for polytopes with large numbers of redundant inequalities or a small number of vertices. However, as there can be exponentially more vertices than there are inequalities, the applicability is limited to polytopes with a small vertex count.

A contour-tracking approach is proposed in [13] in which the skeleton of the projection is traced. The skeleton is formed from the vertices and the one-dimensional lines that join them. The complexity is a linear function of the number of vertices of the projection and as a result is suitable only when the number of vertices is small when compared to the number of inequalities.

An approach, similar to that developed in this paper, was outlined in [1]. The purpose was to compute both the vertex and half-space representation of the projection, although the algorithm can be adapted to compute only the inequalities if desired. Three restrictive assumptions are made, which we relax in this paper. First, in [1] it is assumed that the polytope is in general position; while there exist standard techniques to ensure that this is the case for general polytopes [7], these techniques can cause the description of the projection to increase significantly in size. Second, it is implicitly assumed in [1] that the description of the polytope is irredundant. While this requirement is not necessarily prohibitive and can be satisfied by running as many linear programs as there are constraints, it will greatly slow the algorithm if the dimension is large, or if there are many redundant inequalities. Finally, the assumption is made in [1] that every face of the polytope that projects to a facet of the projection is of dimension $d-1$. While there is a class of polytopes for which this is the case, it is not true in general and we remove this assumption in this paper.

The remainder of this paper is organised as follows. Section 2 provides an introduction to polytopes and their projections and introduces the notion of an equality set. Section 3 gives an outline of the ESP algorithm while Sections 4 through 6 provide the details. The complexity of the algorithm is investigated in Section 7 and numerical simulation results are reported in Section 8. Concluding remarks are made in Section 9. Appendix A details a well-known algorithm for the calculation of the affine hull of a polytope. Appendices B and C show how to modify polytopes that do not contain the origin or are not full dimensional so that their projections can be computed via ESP.

## Notation

If $A \in \mathbb{R}^{m\times n}$ is a matrix, then $\operatorname{null}(A) \triangleq \{x \in \mathbb{R}^n \mid Ax = 0\}$ and $\operatorname{range}(A) \triangleq \{y \in \mathbb{R}^m \mid \exists x \in \mathbb{R}^n,\ y = Ax\}$ are the nullspace and columnspace of $A$ respectively. $\mathsf{N}(A)$ and $\mathsf{R}(A)$ are defined as matrices whose columns form an orthonormal basis for $\operatorname{null}(A)$ and $\operatorname{range}(A)$ respectively. Note that in general $\mathsf{N}(A)$ and $\mathsf{R}(A)$ are not unique. In order to ensure uniqueness, if the rank of $A$ is $r$ and $U\Sigma V^T$ is the singular value decomposition of $A$, then we define $\mathsf{R}(A)$ to be the first $r$ columns of $U$ and $\mathsf{N}(A)$ to be the last $n-r$ columns of $V$. The symbol $^\dagger$ will be used to define the Moore-Penrose pseudo-inverse. If $S$ and $U$ are sets, then $S\backslash U = \{x \mid x \in S,\ x \notin U\}$ is the difference operator and $|S|$ is the cardinality of $S$.

# 2 Polytopes and Projections

This section provides an overview of projections of polytopes. The notion of an equality set will be introduced and the properties relevant to projection will be demonstrated.

## 2.1 Affine Sets and Polytopes

Vector spaces and the related notions of subspaces and affine sets are fundamental to any discussion of polytopes and therefore we begin this introduction with a brief review of these concepts.

A *vector space* $V$ over the reals is a set of vectors $z \in \mathbb{R}^n$ that contains the origin and is closed under vector addition and scalar multiplication. A *subspace* of a vector space $V$ is any subset of $L \subseteq V$ that is itself also a vector space. Recall that $L$ can be represented in two common forms: as the set of all vectors satisfying a finite set of homogeneous linear equations:

$$L = \{z \in \mathbb{R}^n \mid Az = 0\}, \qquad \text{for some } A \in \mathbb{R}^{q \times n} \tag{1}$$

or in terms of the span of a finite set of vectors $v_i \in \mathbb{R}^n$:

$$L = \text{span}\,\{v_1, \dots, v_p\} = \left\{ z \in \mathbb{R}^n \;\middle|\; z = \sum_{i=1}^{p} \lambda_i v_i,\ \lambda_i \in \mathbb{R} \right\}. \tag{2}$$

If the vectors $v_i$ are linearly independent, then the set $\{v_1, \dots, v_p\}$ forms a *basis* for $L$.

The dimension of a subspace $L \subseteq \mathbb{R}^n$, denoted $\dim L$, is defined as the smallest number of vectors whose span is $L$. Note that if the subspace is defined as in (1), then the dimension can be calculated as $\dim L = n - \text{rank}\, A$. If it is represented as in (2), then its dimension is the number of linearly independent vectors $v_i$.

A subset $M \subseteq \mathbb{R}^n$ is called an *affine set* if $(1-\lambda)x + \lambda y \in M$ for every $x$, $y \in M$ and $\lambda \in \mathbb{R}$. Two affine sets $M_1$ and $M_2$ are said to be *parallel* if they can be written as

$$M_1 = M_2 + a, \qquad \text{for some } a \in \mathbb{R}^n. \tag{3}$$

**Theorem 1.** *[14, Thm 1.2] Each non-empty affine set $M$ is parallel to a unique subspace $L$. This $L$ is given by*

$$L = M - M = \{x - y \mid x, y \in M\}.$$

Theorem 1 and (3) allow us to write every affine set $M$ as a *translate* of a unique subspace $L$

$$M = L + a, \qquad \text{for some } a \in \mathbb{R}^n. \tag{4}$$

The two representations of subspaces defined above give rise to two representations of affine sets. Equation 1 gives the representation in terms of all solutions of a finite set of linear equations:

$$\begin{aligned}
M &= L + a, && \text{for some } a \in \mathbb{R}^n \\
&= \{z \in \mathbb{R}^n \mid Az = 0\} + a \\
&= \{z \in \mathbb{R}^n \mid A(z-a) = 0\} \\
&= \{z \in \mathbb{R}^n \mid Az = b\}, && b \triangleq Aa. \qquad (5)
\end{aligned}$$

Affine sets of the form $\left\{z \mid a^T z = b\right\}$, $a \in \mathbb{R}^n$, $b \in \mathbb{R}$ are called *hyperplanes*. Equation 5 shows that all affine sets can be written as the intersection of a finite number of hyperplanes.

Given a set $S \subseteq \mathbb{R}^n$, the *affine hull* of $S$, denoted $\operatorname{aff} S$, is the intersection of all affine sets containing $S$. Clearly, the affine hull of an affine set is itself. It can be shown that the affine hull of $S$ consists of all the vectors $y$ of the form $y = \sum_{i=1}^{p} \lambda_i v_i$, such that $v_i \in S$ and $\sum_{i=1}^{p} \lambda_i = 1$ [14]. Equation 2 allows us to write any affine set as the affine hull of a finite number of vectors:

$$\begin{aligned}
M &= L + a, && \text{for some } a \in \mathbb{R}^n \\
&= \left\{ z \in \mathbb{R}^n \;\middle|\; z = \sum_{i=1}^{p} \lambda_i v_i,\ \lambda_i \in \mathbb{R} \right\} + a \\
&= \left\{ z \in \mathbb{R}^n \;\middle|\; z = a + \sum_{i=1}^{p} \lambda_i v_i,\ \lambda_i \in \mathbb{R} \right\} \\
&= \left\{ z \in \mathbb{R}^n \;\middle|\; z = \lambda_{p+1} a + \sum_{i=1}^{p} \lambda_i (v_i + a),\ \sum_{i=1}^{p+1} \lambda_i = 1,\ \lambda_i \in \mathbb{R} \right\} \\
&= \left\{ z \in \mathbb{R}^n \;\middle|\; z = \sum_{i=1}^{p+1} \lambda_i \mu_i,\ \sum_{i=1}^{p+1} \lambda_i = 1,\ \lambda_i \in \mathbb{R} \right\} \\
&= \operatorname{aff} \{\mu_1, \ldots, \mu_{p+1}\},
\end{aligned}$$

where $\mu_i \triangleq v_i + a,\ i = 1, \ldots, p,\ \mu_{p+1} \triangleq a$.

If $z_1, z_2 \in M$, where $M = L + a$ is an affine set, $L$ is a subspace and $a$ is a vector then $z_1$ and $z_2$ are called *affinely independent* if and only if $z_1 - a$ and $z_2 - a$ are linearly independent. This allows the notion of dimension to be extended to affine sets as follows: the dimension of an affine set $M$, denoted $\dim M$, is defined as the dimension of the subspace which is parallel to it, $\dim M \triangleq \dim L$ if $M = L + a$, where $L$ is a subspace.

A subset $C \subseteq \mathbb{R}^n$ is *convex* if $(1-\lambda)x + \lambda y \in C$ whenever $x, y \in C$ and $0 \leq \lambda \leq 1$. Clearly, all affine sets are convex. The *convex hull* of a set $S \subseteq \mathbb{R}^n$ is the intersection of all the convex sets containing $S$ and is denoted $\operatorname{conv} S$. The *convex combination* of a set of vectors $\{v_1, \ldots, v_p\}$ is all points $\sum_{i=1}^{p} \lambda_i v_i$, $\sum_{i=1}^{p} \lambda_i = 1$, $\lambda_i \geq 0$.

**Theorem 2.** *[14, Thm 2.3] For any $S \subset \mathbb{R}^n$, $\operatorname{conv} S$ consists of all convex combinations of* *the elements of $S$.*

**Corollary 3.** *[14, Cor. 2.4] The convex hull of a finite subset $\{v_1, \ldots, v_p\}$ of $\mathbb{R}^n$ consists of all vectors of the form $\sum_{i=1}^{p} \lambda_i v_i$, with $\lambda_i \geq 0$, $\sum_{i=1}^{p} \lambda_i = 1$.*

A closed halfspace is a convex set of the form $\left\{z \in \mathbb{R}^n \mid a^T z \leq b\right\}$, $a \in \mathbb{R}^n$, $b \in \mathbb{R}$. Any set that can be expressed as the intersection of a finite number of closed halfspaces is called a *polyhedral* (convex) set:

$$P \triangleq \{z \in \mathbb{R}^n \mid Az \leq b\}, \qquad A \in \mathbb{R}^{q \times n},\ b \in \mathbb{R}^q.$$

The dimension of a polyhedron $P$ is the dimension of its affine hull:

$$\dim P \triangleq \dim \operatorname{aff} P.$$

If a halfspace constraint can be removed from the description of $P$ without changing the polyhedron, then it is called *redundant*. If the description of $P$ contains no redundant constraints, then it is called *irredundant*.

Note that while the convex hull of a set of points is always bounded, polyhedra may not be. If a polyhedra is bounded, then it is called a polytope. Throughout the remainder of this paper, we shall assume that all polyhedra are bounded.

The Minkowski-Weyl Theorem (Theorem 4 below) is fundamental as it allows a polytope to be expressed in two forms: as the intersection of a finite number of halfspaces or as the convex hull of a finite number of vectors.

**Theorem 4.** *(Minkowski-Weyl) A subset $P \subset \mathbb{R}^n$ is the convex hull of a finite set of vectors*

$$P = \operatorname{conv}\{v_1, \ldots, v_p\}, \qquad \text{where each } v_i \in \mathbb{R}^n,$$

*if and only if it is a bounded intersection of halfspaces*

$$P = \{z \in \mathbb{R}^n \mid Az \leq b\}, \qquad \text{for some } A \in \mathbb{R}^{q \times n},\ b \in \mathbb{R}^q.$$

Efficient methods exist for converting between one representation and the other [2,8]. However, there can be an exponential relationship between the representations: if a $n-$dimensional polytope can be described by the intersection of $q$ halfspaces, it may take up to $\mathcal{O}(q^{\lfloor \frac{n}{2} \rfloor})$ vertices to formulate the convex hull. Similarly, if a polytope is given as the convex hull of $m$ vertices, an exponential number of halfspaces may be needed to describe the same object. Many applications produce polytopes that exhibit this exponential relationship. The projection method presented here operates only on the polytope in halfspace form without computing the vertex representation and so is particularly suited to polytopes that require a

Table 1: Face names of an $n$-dimensional polytope

| Dimension | Name |
|---|---|
| $(n-1)$-face | Facet |
| $(n-2)$-face | Ridge |
| 1-face | Edge |
| 0-face | Vertex |

small number of halfspaces to describe them but a very large number of points to formulate as a convex hull.

### 2.2 Face Lattice

In this section we recall a natural decomposition of polytopes: the face lattice.

**Definition 5.** *(Face) [19, Def. 2.1] $F$ is a* face *of the polytope $P \subset \mathbb{R}^n$ if there exists a hyperplane $\left\{z \in \mathbb{R}^n \mid a^T z = b\right\}$, where $a \in \mathbb{R}^n$, $b \in \mathbb{R}$, such that*

$$F = P \cap \left\{z \in \mathbb{R}^n \mid a^T z = b\right\}$$

*and $a^T z \leq b$ for all $z \in P$.*

Note that $\emptyset$ and $P$ are both faces of $P$ (consider the hyperplanes $\left\{z \mid 0^T z = 1\right\}$ and $\left\{z \mid 0^T z = 0\right\}$, respectively); all other faces are called *proper* faces.

**Theorem 6.** *[19, Prop. 2.2] Let $P \subset \mathbb{R}^n$ be a polytope and let $F$ be a face of $P$:*

1. *The face $F$ is a polytope.*
2. *Every intersection of faces of $P$ is a face of $P$.*
3. *The faces of $F$ are exactly the faces of $P$ that are contained in $F$.*
4. *$F = P \cap \operatorname{aff} F$.*

Note that as each face of a polytope is itself a polytope, the notion of dimensionality applies to faces. If a face is of dimension $p$, then it is called a $p$-face. Faces of particular dimensions have explicit names as shown in Table 1. Note that for a polytope in $\mathbb{R}^n$, the word *face* refers to a face of arbitrary dimension, while *facet* refers specifically to the $(n-1)$-faces.

Theorem 6 allows a natural partial ordering over the faces of a polytope based on inclusion. If $P$ is a polytope, then each face of $P$ is itself a polytope whose faces are a subset of $P$'s faces. This allows us to draw a *face lattice* where each node represents a face of the polytope and two nodes are connected if one is a subset of the other. The lattice is organized vertically, such that faces of the same dimension appear at the same level and the dimension increases up the diagram. An example lattice of a cube is shown in Figure 1.

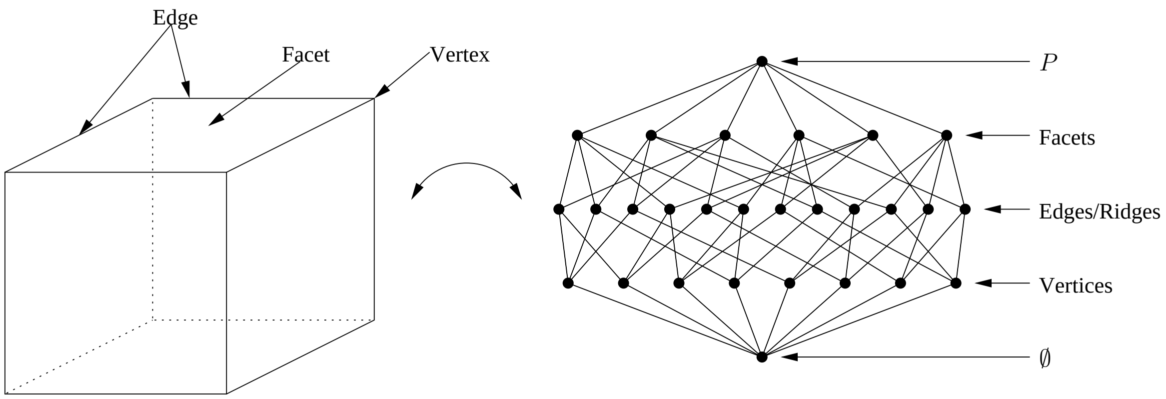



Figure 1: Face lattice of a cube

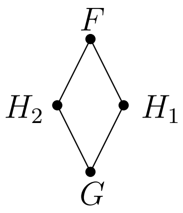



Figure 2: Illustration of "Diamond Property"

A property of the face lattice that is key to the projection method developed here is the so-called *diamond property*:

**Theorem 7.** (*Diamond property) [19, Thm 2.7(iii)] If $G$ and $F$ are faces of a polytope $P$ and $G \subset F$ with* $\dim F - \dim G = 2$, *then there are exactly two faces $H_1$, $H_2$ with the property $G \subset H_1 \subset F$ and $G \subset H_2 \subset F$.*

Note that, as the name of the proposition implies, the four faces in Theorem 7 will "look like" a diamond, as shown in Figure 2.

The immediate implication of Theorem 7 is that any two facets ($(n-1)$-faces) of a polytope are either not joined in the face lattice or are connected by the inclusion of exactly one ridge.

Two faces $F_1$ and $F_2$ of $P$ are said to be *adjacent* if the intersection $F_1 \cap F_2$ is a facet of both.

### 2.3 Equality Sets of a Polytope

We now introduce the notion of the *equality set*, which is the method that will be used in this paper for defining faces.

If $P \triangleq \{z \in \mathbb{R}^n \mid Az \leq b\}$ is a polytope defined by the intersection of $q$ halfspaces and $\mathsf{E} \subseteq \{1, \ldots, q\}$, then $A_{\mathsf{E}}$ is a matrix whose rows are the rows of $A$ whose indices are in $\mathsf{E}$.

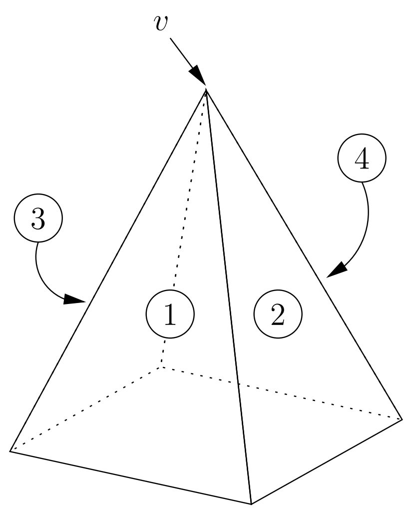



Figure 3: Illustration of equality sets

Similarly, $b_{\mathsf{E}}$ is the vector formed by the rows of $b$ whose indices are in $\mathsf{E}$. The relevant rows of $A_{\mathsf{E}}$ are taken in the same order as those of the matrix $A$. If $\mathsf{E} = \{i\}$ is a singleton, then we write $A_i$ for $A_{\{i\}}$. The notation $P_{\mathsf{E}}$ refers to the set $P_{\mathsf{E}} \triangleq P \cap \{z \mid A_{\mathsf{E}} z = b_{\mathsf{E}}\}$.

**Definition 8.** *(Equality Set) Let $P \triangleq \{z \mid Az \leq b\}$ be a polytope defined by the intersection of $q$ halfspaces, $\mathsf{E} \subseteq \{1, \ldots, q\}$ and*

$$G(\mathsf{E}) \triangleq \{i \in \{1, \ldots, q\} \mid A_i z = b_i,\ \forall z \in P_{\mathsf{E}}\}.$$

*The set $\mathsf{E}$ is an* equality set *of $P$ if and only if $\mathsf{E} = G(\mathsf{E})$.*

**Remark 9.** *Note that this definition of an equality set is similar to the notion of an* equality subsystem *used in [4]. Whereas equality subsystems generally refer to any description of the affine set that describes the affine hull of the face, equality sets describe this affine set specifically in terms of* all *of the inequalities of $P$ that are met with equality at all points in the face.*

To illustrate the idea of an equality set, consider the pyramid shown in Figure 3. Notice that the set $P_{\{1,2,3\}}$ is the face, or more specifically, the vertex $v$, but that $\{1, 2, 3\}$ is not an equality set, since $G(\{1, 2, 3\}) = \{1, 2, 3, 4\}$. There is only one equality set that defines the face $v$, and that is $\{1, 2, 3, 4\}$.

The goal is to re-write the face lattice in terms of equality sets. First, the relationship between equality sets and faces must be made explicit: there is a one-to-one mapping between faces and equality sets.

We begin with the following well-known result: equality sets define affine hulls.

**Lemma 10.** *If* $\mathsf{E}$ *is an equality set of the polytope* $P \triangleq \{z \in \mathbb{R}^n \mid Az \le b\}$*, then* $\operatorname{aff} P_{\mathsf{E}} = \{z \in \mathbb{R}^n \mid A_{\mathsf{E}} z = b_{\mathsf{E}}\}$ *and* $\dim P_{\mathsf{E}} = n - \operatorname{rank} A_{\mathsf{E}}$*.*

*Proof.* The first result follows directly from Definition 8. The second result follows from $\dim P_{\mathsf{E}} = \dim \operatorname{aff} P_{\mathsf{E}} = n - \operatorname{rank} A_{\mathsf{E}}$. □

**Lemma 11.** *(Uniqueness of Equality Sets) If* $\mathsf{E}$ *and* $\mathsf{B}$ *are equality sets of a polytope* $P \triangleq \{z \in \mathbb{R}^n \mid Az \le b\}$*, then* $\mathsf{E} = \mathsf{B}$ *if and only if* $P_{\mathsf{E}} = P_{\mathsf{B}}$*.*

*Proof.* Clearly, if $\mathsf{E} = \mathsf{B}$, then $P_{\mathsf{E}} = P_{\mathsf{B}}$.

Assume that $P_{\mathsf{E}} = P_{\mathsf{B}}$. Since $\mathsf{E}$ and $\mathsf{B}$ are equality sets, $\mathsf{E} = G(\mathsf{E})$ and $\mathsf{B} = G(\mathsf{B})$ and because $P_{\mathsf{E}} = P_{\mathsf{B}}$ we have

$$\begin{aligned}
\mathsf{E} = G(\mathsf{E}) &= \{i \mid A_i z = b_i,\ \forall z \in P_{\mathsf{E}}\} \\
&= \{i \mid A_i z = b_i,\ \forall z \in P_{\mathsf{B}}\} \\
&= G(\mathsf{B}) = \mathsf{B}.
\end{aligned}$$

□

Theorem 12 proves the assertion that there is a one-to-one mapping between equality sets and faces.

**Theorem 12.** *If* $\mathsf{E}$ *is an equality set of the polytope* $P \triangleq \{z \in \mathbb{R}^n \mid Az \le b\}$*, then* $P_{\mathsf{E}}$ *is a face of* $P$*. Furthermore, if* $F$ *is a face of* $P$*, then there exists a unique equality set* $\mathsf{E}$ *such that* $F = P_{\mathsf{E}}$*.*

*Proof.* If $i \in \mathsf{E}$, then $A_i z \le b_i$ is true for all $z \in P$ and therefore $P \cap \{z \in \mathbb{R}^n \mid A_i z = b_i\}$ is a face by Definition 5. By Proposition 6(2), all intersections of faces are faces and thus $P_{\mathsf{E}} = P \cap_{i \in \mathsf{E}} \{z \in \mathbb{R}^n \mid A_i z = b_i\}$ is a face of $P$.

Recall from Proposition 6(1) that every face $F$ of $P$ is a polytope. The affine hull of a polytope can be represented by the intersection of all halfspaces that are satisfied with equality at all points in the polytope. Let the indices of those halfspaces be $\mathsf{E}$; clearly $\mathsf{E} = G(\mathsf{E})$ and $\mathsf{E}$ is an equality set. By Proposition 6(4) and using Lemma 10, $F$ can be written as

$$\begin{aligned}
F &= P \cap \operatorname{aff} F \\
&= P \cap \operatorname{aff} \{z \mid A_{\mathsf{E}} z = b_{\mathsf{E}}\} \\
&= P \cap \operatorname{aff} P_{\mathsf{E}} \\
&= P_{\mathsf{E}}.
\end{aligned}$$

□

Theorem 13 allows the ordering of the face lattice to be written as set inclusion on equality sets.

**Theorem 13.** *Let* $\mathsf{E}$ *and* $\mathsf{B}$ *be equality sets of* $P$*. The inclusion* $P_{\mathsf{E}} \subset P_{\mathsf{B}}$ *holds if and only if* $\mathsf{E} \supset \mathsf{B}$*.*

*Proof.* Let the polytope be defined by the matrix $A \in \mathbb{R}^{q\times n}$ and the vector $b \in \mathbb{R}^q$: $P \triangleq \{x \mid Ax \le b\}$.

If $\mathsf{E} \supset \mathsf{B}$, we can write $\mathsf{E} = \mathsf{B} \cup X$ for some $X \subseteq \{1, \dots, q\}$ with $X \cap \mathsf{B} = \emptyset$ and therefore

$$\begin{aligned} P_{\mathsf{E}} &= P_{\mathsf{B}\cup X} \\ &= P_{\mathsf{B}} \cap P_X \\ &\subseteq P_{\mathsf{B}}. \end{aligned}$$

We now show that there exists an $i$ in $X$ that is not in $B$ and therefore we have strict inclusion. If $i$ is in $X$ then it is not in $B$ because $X \cap \mathsf{B} = \emptyset$ and therefore by the definition of equality set, there exists a $z \in P_{\mathsf{B}}$ such that $A_i z \ne b_i$, which implies that $z \notin P_X$ and hence we have strict inclusion, i.e. $P_{\mathsf{E}} \subset P_{\mathsf{B}}$.

Assume that $P_{\mathsf{E}} \subset P_{\mathsf{B}}$. If $i$ is in the equality set $\mathsf{B}$, then for all $z$ in $P_{\mathsf{E}}$, we have that $A_i z = b_i$ because every point in $P_{\mathsf{E}}$ must satisfy all constraints in $P_{\mathsf{B}}$. By definition, an equality set contains all constraints that are satisfied with equality at all points in the polytope and therefore $i$ must be in the equality set $\mathsf{E}$. It follows directly that $\mathsf{E} \supset \mathsf{B}$.

□

Theorems 12 and 13 allow the face lattice to be written in terms of equality sets. The example of the cube lattice is shown again in Figure 4, but now in terms of equality sets.

## 2.4 Projection of a Polytope

We now turn our attention to the main topic of this paper: projection.

**Definition 14.** *If* $P \subset \mathbb{R}^d \times \mathbb{R}^k$ *is a polytope then the* projection *of* $P$ *onto* $\mathbb{R}^d$ *is* $\pi_d(P) \triangleq \left\{x \in \mathbb{R}^d \mid \exists y \in \mathbb{R}^k,\ (x, y) \in P\right\}$.

Note that this definition of projection is also referred to as "projection along the coordinate axes" or "orthogonal projection" in the literature.

The inputs to the algorithm presented in this report are the matrices $C \in \mathbb{R}^{q\times d}$ and $D \in \mathbb{R}^{q\times k}$ and the vector $b \in \mathbb{R}^q$, which define the polytope $P \triangleq \left\{(x, y) \in \mathbb{R}^d \times \mathbb{R}^k \mid Cx + Dy \le b\right\}$. The goal is to compute a matrix $G \in \mathbb{R}^{p\times d}$ and a vector $g \in \mathbb{R}^p$ that define the projection of $P$ onto $\mathbb{R}^d$, $\pi_d(P) = \left\{x \in \mathbb{R}^d \mid Gx \le g\right\}$. Proposition 15 states that the projection of a polytope is a polytope, and hence can be expressed in the required form, as the intersection of a finite number of halfspaces.

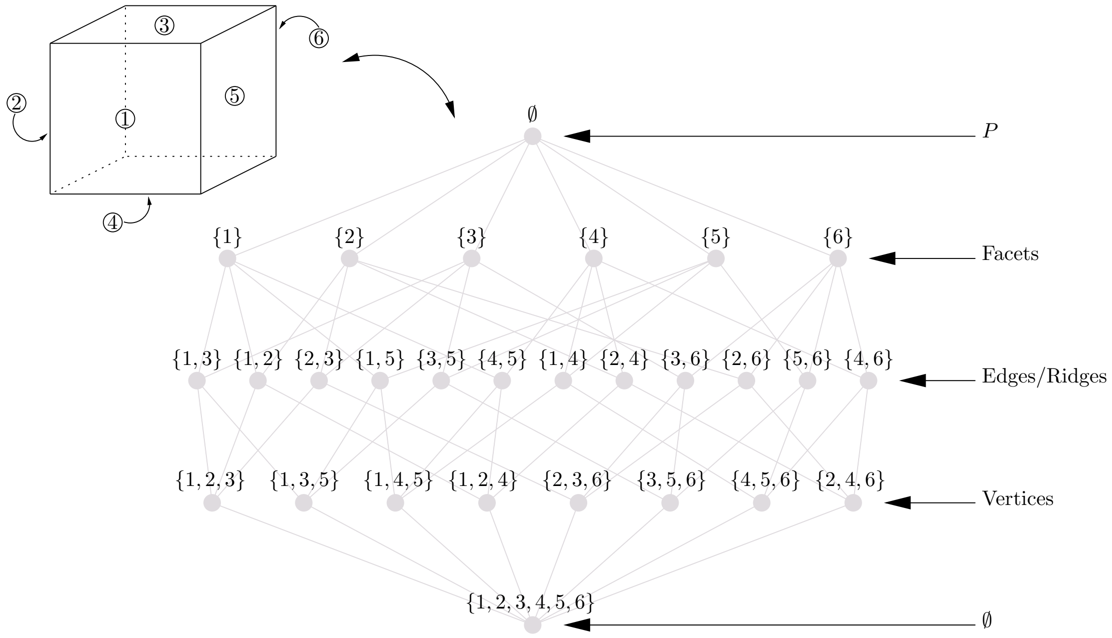



Figure 4: Equality set lattice of a cube

**Proposition 15.** *[19] If $P \subset \mathbb{R}^d \times \mathbb{R}^k$ is a polytope, then the projection of $P$ onto $\mathbb{R}^d$ is a polytope.*

Proposition 16 allows the calculation of the dimension of a face of $P$, given its defining equality set.

**Proposition 16.** *[4] If $\mathsf{E}$ is an equality set of the polytope $P = \left\{(x, y) \in \mathbb{R}^d \times \mathbb{R}^k \mid Cx + Dy \leq b\right\}$, then*

$$\begin{aligned} \dim \pi_d\left(P_{\mathsf{E}}\right) &= \dim P_{\mathsf{E}} - k + \operatorname{rank} D_{\mathsf{E}} \\ &= d + \operatorname{rank} D_{\mathsf{E}} - \operatorname{rank} \begin{bmatrix} C_{\mathsf{E}} & D_{\mathsf{E}} \end{bmatrix} \end{aligned}$$

Proposition 17 implies that if the affine hulls of each of the facets of the projection can be found, and expressed in the form of linear equations, then we can directly find an appropriate matrix $G$ and a vector $g$ such that $\pi_d(P) = \left\{x \in \mathbb{R}^d \mid Gx \leq g\right\}$.

**Proposition 17.** *[18, Theorem 3.2.1] If $F_1, \ldots, F_t$ are the facets of a polytope $P \in \mathbb{R}^n$, $0 \in P$ and $\operatorname{aff} F_i = \left\{x \in \mathbb{R}^n \mid a_i^T x = b_i\right\}$, where each $a_i \in \mathbb{R}^n, b_i \in \mathbb{R}$, then*

$$P = \operatorname{aff} P \cap \left\{ x \in \mathbb{R}^n \;\middle|\; \begin{bmatrix} a_1^T \\ \vdots \\ a_t^T \end{bmatrix} x \leq \begin{bmatrix} b_1 \\ \vdots \\ b_t \end{bmatrix} \right\},$$

*where the sign of $b_i$ is chosen so that $b_i \geq 0$ for all $i$.*

We now develop the theory required to compute the affine hulls of the facets of the projection as a function of the polytope $P$. Proposition 18 and the related Corollary 19, show that the faces of the projection $\pi_d(P)$ are projections of faces of the polytope $P$, which can be expressed in terms of the equality sets of $P$.

**Proposition 18.** *[19, Lem. 7.10] If $P \subset \mathbb{R}^d \times \mathbb{R}^k$ is a polytope, then for every face $F$ of $\pi_d(P)$, the preimage $\pi_d^{-1}(F) = \{y \in P \mid \pi_d(y) \in F\}$ is a face of $P$.*

*Furthermore, if $F$ and $G$ are faces of $\pi_d(P)$, then $F \subset G$ holds if and only if $\pi_d^{-1}(F) \subset \pi_d^{-1}(G)$.*

**Corollary 19.** *If $P \subset \mathbb{R}^d \times \mathbb{R}^k$ is a polytope defined by the intersection of $q$ halfspaces, then for every face $F$ of $\pi_d(P)$, there exists a unique equality set $\mathsf{E} \subseteq \{1, \ldots, q\}$ of $P$ such that the preimage $\pi_d^{-1}(F) = P_{\mathsf{E}}$.*

*Furthermore, if $F$ and $G$ are faces of $\pi_d(P)$ such that $\pi_d^{-1}(F) = P_{\mathsf{E}}$ and $\pi_d^{-1}(G) = P_{\mathsf{B}}$, where $\mathsf{E}$ and $\mathsf{B}$ are equality sets of $P$, then $F \subset G$ holds if and only if $\mathsf{E} \supset \mathsf{B}$.*

*Proof.* Proposition 12 states that every face can be expressed as $P_{\mathsf{E}}$ for a unique equality set and therefore the first result follows from Proposition 18. The second result follows directly from Theorem 13 and Proposition 18. □

Proposition 20 shows that the projection of an affine hull is the affine hull of the projection.

**Proposition 20.** *If $P \subset \mathbb{R}^d \times \mathbb{R}^k$ is a polytope then* $\operatorname{aff} \pi_d(P) = \pi_d(\operatorname{aff} P)$.

*Proof.* We first show the inclusion $\operatorname{aff} \pi_d(P) \subseteq \pi_d(\operatorname{aff} P)$. If $\dim \pi_d(P) = n$, then $\operatorname{aff} \pi_d(P) = \operatorname{aff} \{x_1, \ldots, x_{n+1}\}$, for some $n+1$ affinely independent points, $x_i \in \pi_d(P)$. Since $\pi_d(P)$ is the projection of $P$, for each $x_i$, there exists a $y_i \in \mathbb{R}^k$ such that $(x_i, y_i) \in P$, and therefore $x_i \in \{x \mid \exists y,\ (x, y) \in \operatorname{aff} P\} = \pi_d(\operatorname{aff} P)$. It follows directly that the affine hull of $\{x_1, \ldots, x_{n+1}\}$ is a subset of the projection of the affine hull of $P$,

$$\operatorname{aff} \pi_d(P) \subseteq \pi_d(\operatorname{aff} P).$$

We now show the inclusion $\operatorname{aff} \pi_d(P) \supseteq \pi_d(\operatorname{aff} P)$. If $\dim P = m$, then $\operatorname{aff} P = \operatorname{aff} \{(\xi_1, \upsilon_1), \ldots, (\xi_{m+1}, \upsilon_{m+1})\}$, for some $m+1$ affinely independent points $(\xi_i, \upsilon_i) \in$

$P$. Hence,

$$\begin{aligned}
\pi_d(\text{aff } P) &= \pi_d\left(\text{aff } \{(\xi_1, \upsilon_1), \ldots, (\xi_{m+1}, \upsilon_{m+1})\}\right) \\
&= \text{aff}\left\{ x \;\middle|\; \exists y, \begin{bmatrix} x \\ y \end{bmatrix} = \sum_{i=1}^{m+1} \lambda_i \begin{bmatrix} \xi_i \\ \upsilon_i \end{bmatrix}, \; \sum_{i=1}^{m+1} \lambda_i = 1 \right\} \\
&= \text{aff}\left\{ x \;\middle|\; x = \sum_{i=1}^{m+1} \lambda_i \xi_i, \; \sum_{i=1}^{m+1} \lambda_i = 1, \; \exists y, \; y = \sum_{i=1}^{m+1} \lambda_i \upsilon_i \right\} \\
&= \text{aff } \{\xi_1, \ldots, \xi_{m+1}\} \\
&\subseteq \text{aff } \pi_d(P),
\end{aligned}$$

where the inclusion follows because $\xi_i \in \pi_d(P)$ for all $i$.

□

Recall from Lemma 10 that equality sets define affine hulls, i.e., $\text{aff } P_{\mathsf{E}} = \{(x, y) \mid C_{\mathsf{E}} x + D_{\mathsf{E}} y = b_{\mathsf{E}}\}$, if $P \triangleq \{(x, y) \mid Cx + Dy \leq b\}$ and $\mathsf{E}$ is an equality set. If $\mathsf{E}$ also has the property that $\pi_d(P_{\mathsf{E}})$ is a facet of $\pi_d(P)$, then Proposition 20 allows the affine hull of the facet to be written as

$$\begin{aligned}
\text{aff } \pi_d(P_{\mathsf{E}}) &= \pi_d(\text{aff } P_{\mathsf{E}}) \\
&= \pi_d\left(\text{aff } \{(x, y) \mid C_{\mathsf{E}} x + D_{\mathsf{E}} y = b_{\mathsf{E}}\}\right)
\end{aligned}$$

Proposition 21 then provides a row of the matrix $G$ and an element of the vector $g$:

$$\text{aff } \pi_d(P_{\mathsf{E}}) = \left\{ x \;\middle|\; \mathsf{N}\left({D_{\mathsf{E}}}^T\right)^T C_{\mathsf{E}} x = \mathsf{N}\left({D_{\mathsf{E}}}^T\right)^T b_{\mathsf{E}} \right\}.$$

**Proposition 21.** *If $M \triangleq \left\{(x, y) \in \mathbb{R}^d \times \mathbb{R}^k \mid C_{\mathsf{E}} x + D_{\mathsf{E}} y = b_{\mathsf{E}}\right\}$ is an affine set, then the projection of $M$ onto $\mathbb{R}^d$ is*

$$\pi_d(M) = \left\{ x \in \mathbb{R}^d \;\middle|\; \mathsf{N}\left({D_{\mathsf{E}}}^T\right)^T C_{\mathsf{E}} x = \mathsf{N}\left({D_{\mathsf{E}}}^T\right)^T b_{\mathsf{E}} \right\}.$$

*Proof.* This statement can be derived directly as follows:

$$\begin{aligned}
\pi_d(M) &= \left\{ x \in \mathbb{R}^d \;\middle|\; \exists y \in \mathbb{R}^k, \; C_{\mathsf{E}} x + D_{\mathsf{E}} y = b_{\mathsf{E}} \right\} \\
&= \left\{ x \in \mathbb{R}^d \mid C_{\mathsf{E}} x - b_{\mathsf{E}} \in \text{range}(D_{\mathsf{E}}) \right\} \\
&= \left\{ x \in \mathbb{R}^d \;\middle|\; \mathsf{N}\left({D_{\mathsf{E}}}^T\right)^T (C_{\mathsf{E}} x - b_{\mathsf{E}}) = 0 \right\} \qquad (6) \\
&= \left\{ x \in \mathbb{R}^d \;\middle|\; \mathsf{N}\left({D_{\mathsf{E}}}^T\right)^T C_{\mathsf{E}} x = \mathsf{N}\left({D_{\mathsf{E}}}^T\right)^T b_{\mathsf{E}} \right\},
\end{aligned}$$

where (6) follows because the column space of $D_{\mathsf{E}}$ is orthogonal to the left nullspace of $D_{\mathsf{E}}$.

□

The tools are now all in place to compute a matrix $G$ and a vector $g$ such that $\pi_d(P) = \{x \in \mathbb{R}^d \mid Gx \leq g\}$, if all of the equality sets of the polytope $P$, which define faces $P_{\mathsf{E}}$ that project to facets of $\pi_d(P)$, can be found. The algorithm that is presented in the remainder of this report is a search method that will find precisely these sets.

# 3 Algorithm Outline

The Equality Set Projection (ESP) algorithm computes the projection of polytopes that are described in halfspace form and also expresses the projection in the same form. It is an output sensitive algorithm, with a constant number of linear programs required per facet of the projection in the absence of degeneracy. It is therefore most suited for low facet-count, high vertex-count polytopes. This section will outline the basic procedure for computing the projection, while those following will present the details.

The input to the algorithm is the (bounded) polytope $P$, which is described by the intersection of $q$ halfspaces. The data given to the algorithm are the matrices $C \in \mathbb{R}^{q \times d}$, $D \in \mathbb{R}^{q \times k}$ and $b \in \mathbb{R}^q$ that describe $P$ as follows:

$$P \triangleq \left\{(x, y) \in \mathbb{R}^d \times \mathbb{R}^k \mid Cx + Dy \leq b\right\}.$$

The assumption is not made that $P$ is irredundant, although the algorithm will proceed faster if it is. The goal is to compute the matrix $G$ and the vector $g$ such that the axis-aligned projection of $P$ onto the first $d$ coordinates is given by the irredundant description

$$\begin{aligned}\pi_d(P) &\triangleq \left\{x \in \mathbb{R}^d \mid \exists y,\ (x, y) \in P\right\}\\ &= \left\{x \in \mathbb{R}^d \mid Gx \leq g\right\}.\end{aligned}$$

The assumption is made that $\pi_d(P)$ is full-dimensional in $\mathbb{R}^d$ and that the origin is in its interior. Note that this is guaranteed to be true if $P$ is full-dimensional in $\mathbb{R}^d \times \mathbb{R}^k$ and the origin is interior to it. The remainder of this section discusses the algorithm under these assumptions and Appendices B and C show how to apply the algorithm when a polytope does not satisfy these requirements.

As discussed in Section 2, the matrix $G$ and the vector $g$ are formed from the affine hulls of the facets of the projection. Each facet $F$ of the projection $\pi_d(P)$ is defined by a unique equality set $\mathsf{E}$ of $P$ such that $F = \pi_d(P_{\mathsf{E}})$. Therefore, the algorithm is a search procedure for finding these unique equality sets.

The ESP search procedure exploits the relationship between facets and ridges that is captured by the Diamond Property (Theorem 7). The algorithm is initialized by discovering a random facet $F$ of $\pi_d(P)$ (Shooting Oracle). The equality sets that define faces of $P$ that project to the ridges of $\pi_d(P)$ that are subsets of $F$ can then be computed (Ridge Oracle).

A list is maintained that has one element for every ridge discovered. This element consists of two equality sets of $P$ that define a ridge of the projection and one of its containing facets. In each iteration, a ridge is selected from the list, along with its containing facet. Recall that by the Diamond Property, each ridge is contained in exactly two facets. Therefore, given the selected ridge-facet list element, the equality set of the second containing facet can be computed (Adjacency Oracle). As before, the equality sets of all ridges of this new facet are computed and inserted into the list.

Note that because of the Diamond Property, each ridge is visited exactly twice. Therefore, by removing every ridge that appears in the list twice at the end of each iteration, we can guarantee that the algorithm has no cycles and has a finite execution time. The iterative step is repeated until the list is empty, at which time all facets will have been found.

Once all of the equality sets that define faces of $P$ that project to facets of $\pi_d(P)$ have been collected, the matrix $G$ and the vector $g$ can be computed as shown in Propositions 17 and 21. Theorem 22 proves that this procedure finds all facets of the projection and that it terminates in finite time. It will be seen in the following sections that this statement can be strengthened to polynomial time in the absence of degeneracy.

**Theorem 22.** *The ESP algorithm returns all facets of the projection and terminates in finite time.*

*Proof.* Beginning from a random facet $F$ of $\pi_d(P)$, the ESP algorithm iteratively computes all adjacent facets without re-visiting any facet. Every polytope has a finite number of faces [14, Thm. 19.1] and therefore ESP terminates in finite time.

It remains to be shown that all facets are visited. Given a facet $F_0$, we define the set $\mathcal{F}(F_0)$ to be all facets $F_n$ such that there exists a sequence of facets from $F_0$ to $F_n$ and $F_i$ is adjacent to $F_{i+1},\ i = 0, \ldots, n-1$. Given an initial starting facet $F_0$, the ESP algorithm clearly computes $\mathcal{F}(F_0)$. Therefore, if $\mathcal{F}(F_0)$ covers all facets for any choice of $F_0$, then the ESP algorithm will return all facets of the projection.

We define a graph $H(\mathcal{F}(F_0))$ whose vertices represent the facets of $\mathcal{F}(F_0)$ and two vertices of the graph are connected by an edge if the corresponding facets are adjacent. [19, Cor. 2.14] states that there exists a polytope $\tilde{P}$ such that $H(\mathcal{F}(F_0))$ is combinatorially equivalent to the graph $G(\tilde{P})$ formed from the vertices and edges of $\tilde{P}$. Balinski's Theorem [5] states that $G(\tilde{P})$ is connected and therefore it follows that $H(\mathcal{F}(F_0))$ is connected and the ESP algorithm returns all facets of the projection.

□

Three oracles are needed for this search procedure:

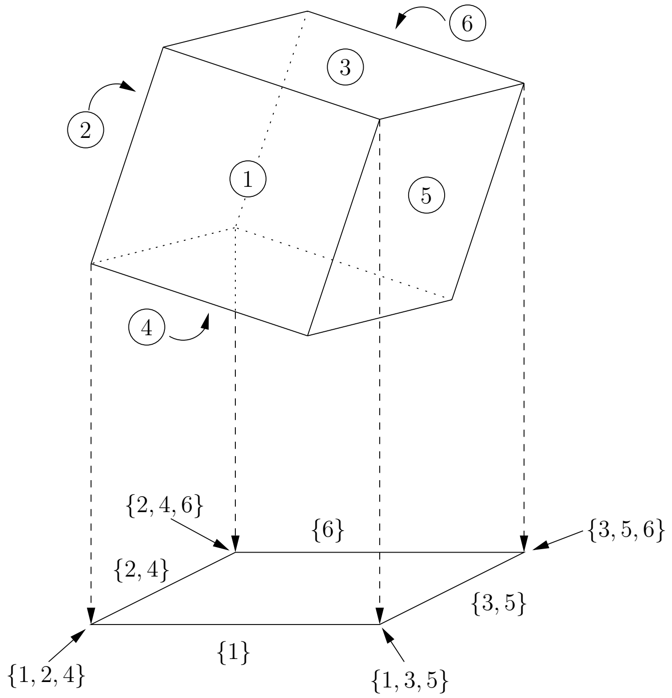



Figure 5: Example projection of a cube

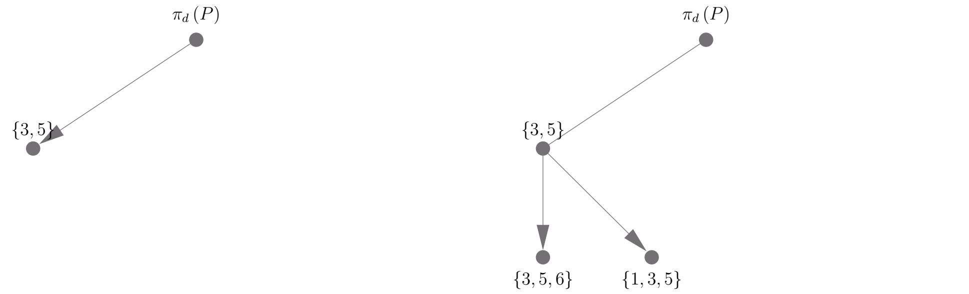
**Adjacency Oracle** $(\mathsf{E}^{\mathsf{adj}}, a_{\mathsf{adj}}, b_{\mathsf{adj}}) = ADJ((\mathsf{E}_r, a_r, b_r), (\mathsf{E}, a_f, b_f))$ This oracle takes two equality sets $\mathsf{E_r}$ and $\mathsf{E}$ of $P$ where $\pi_d\left(P_{\mathsf{E}}\right) = \left\{x \,\middle|\, a_f^T x = b_f\right\} \cap \pi_d\left(P\right)$ is a facet of $\pi_d\left(P\right)$ and $\pi_d\left(P_{\mathsf{E_r}}\right) = \left\{x \,\middle|\, a_r^T x = b_r\right\} \cap \pi_d\left(P_{\mathsf{E}}\right)$ is a ridge with $\pi_d\left(P_{\mathsf{E}}\right) \supset \pi_d\left(P_{\mathsf{E_r}}\right)$. The oracle returns the unique equality set $\mathsf{E}^{\mathsf{adj}}$ such that $\pi_d\left(P_{\mathsf{E}^{\mathsf{adj}}}\right) \supset \pi_d\left(P_{\mathsf{E_r}}\right),\ \mathsf{E}^{\mathsf{adj}} \neq \mathsf{E}$.

**Ridge Oracle** $(\mathsf{E}_r^1, \ldots, \mathsf{E}_r^m) = RDG(\mathsf{E}, a_f, b_f)$ This oracle takes an equality set $\mathsf{E}$ of $P$ where $\pi_d\left(P_{\mathsf{E}}\right) = \left\{x \,\middle|\, a_f^T x = b_f\right\} \cap \pi_d\left(P\right)$ is a facet of $\pi_d\left(P\right)$ and returns all equality sets $\mathsf{E}_r^i$ of $P$ such that $\pi_d\left(P_{\mathsf{E}_r^i}\right)$ is a ridge of $\pi_d\left(P\right)$ and $\pi_d\left(P_{\mathsf{E}_r^i}\right) \subset \pi_d\left(P_{\mathsf{E}}\right)$.

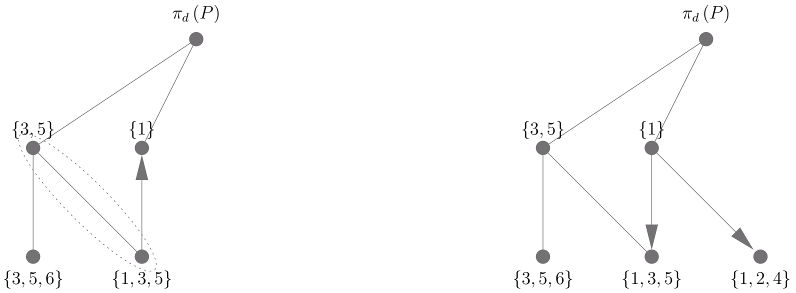
**Ray-shooting Oracle** $(\mathsf{E}_0, a_f, b_f) = SHOOT(P)$ This oracle returns a random equality set $\mathsf{E}_0$ of $P$ such that $\pi_d\left(P_{\mathsf{E}_0}\right) = \left\{x \,\middle|\, a_f^T x = b_f\right\} \cap \pi_d\left(P\right)$ is a facet of the projection.

These oracles are discussed in Sections 4, 5 and 6 respectively. ESP is described in procedural form in Algorithm 3.1.

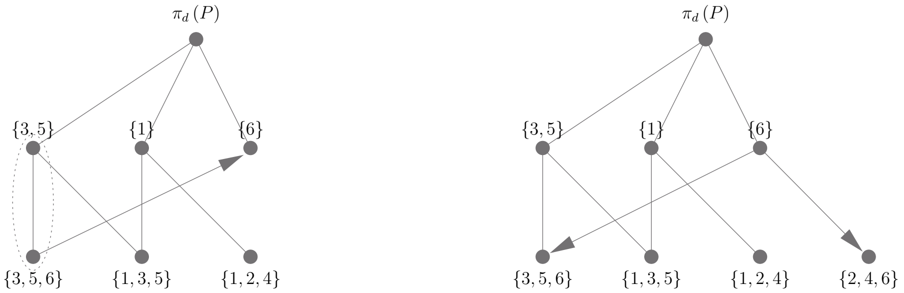
The algorithm is illustrated by the example of a cube in $\mathbb{R}^3$ being projected to $\mathbb{R}^2$ as shown in Figure 5. The sequence shown in Figure 6 demonstrates the construction of the top two levels of the lattice of the projection as the algorithm proceeds. The path that the algorithm takes through the 3D cube lattice is shown in Figure 7.



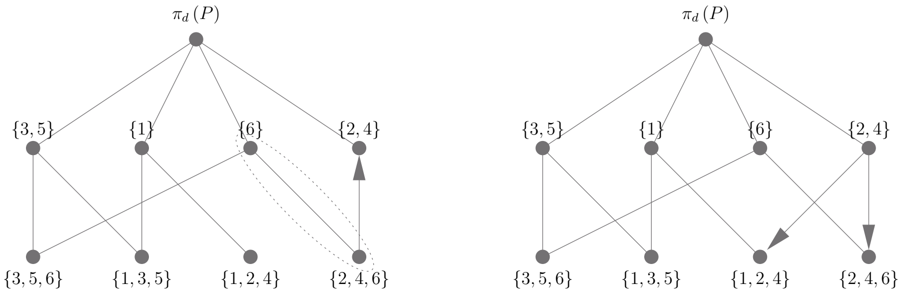

Step 1: $\{3,5\} = SHOOT$ Step 2: $(\{3,5,6\},\{1,3,5\}) = RDG(\{3,5\})$
$List : \mathbf{L} = ((\{3,5,6\},\{3,5\}),(\{1,3,5\},\{3,5\}))$




Step 3: $\{1\} = ADJ(\{1,3,5\},\{3,5\})$ Step 4: $(\{1,3,5\},(1,2,4)) = RDG(\{1\})$
$List : \mathbf{L} = ((\{3,5,6\},\{3,5\}), \cancel{(\{1,3,5\},\{3,5\})}, \cancel{(\{1,3,5\},\{1\})}, (\{1,2,4\},\{1\}))$




Step 5: $\{6\} = ADJ(\{3,5\},\{3,5,6\})$ Step 6: $(\{3,5,6\},\{2,4,6\}) = RDG(\{6\})$
$List : \mathbf{L} = (\cancel{(\{3,5,6\},\{3,5\})}, (\{1,2,4\},\{1\}), \cancel{(\{3,5,6\},\{6\})}, (\{2,4,6\},\{6\})),$




Step 7: $\{2,4\} = ADJ(\{2,4,6\},\{6\})$ Step 8: $(\{1,2,4\},\{2,4,6\}) = RDG(\{2,4\})$
$List : \mathbf{L} = (\cancel{(\{1,2,4\},\{1\})}, \cancel{(\{2,4,6\},\{6\})}, \cancel{(\{1,2,4\},\{2,4\})}, \cancel{(\{2,4,6\},\{2,4\})}) = \emptyset$

Figure 6: ESP procedure: Projection of a cube

**Algorithm 3.1** Equality Set Projection ESP

**Input:** Polytope $P \triangleq \left\{(x,y) \in \mathbb{R}^d \times \mathbb{R}^k \,\middle|\, Cx + Dy \leq b\right\}$ whose projection is full-dimensional and contains the origin in its interior.

**Output:** Matrix $G$ and vector $g$ such that $\left\{x \in \mathbb{R}^d \mid Gx \leq g\right\}$ is an irredundant description of $\pi_d(P)$.
List $\mathbf{E}$ of all equality sets $\mathsf{E}$ of $P$ such that $\pi_d(P_{\mathsf{E}})$ is a facet of $\pi_d(P)$.

***Initialize ridge-facet list L with random facet.***

$\mathbf{L} \longleftarrow \emptyset$
$(\mathsf{E}_0, a_f, b_f) = SHOOT(P)$ Section 6
$\mathbf{E_r} = RDG(\mathsf{E}_0, a_f, b_f)$ Section 5
**for each** element $(\mathsf{E_r}, a_r, b_r)$ in list $\mathbf{E_r}$ **do**
  Add element $((\mathsf{E}_r, a_r, b_r), (\mathsf{E}_0, a_f, b_f))$ to list $\mathbf{L}$
**end for**

***Initialize matrix $G$, vector $g$ and list E.***

$G \longleftarrow a_f^T$, $g \longleftarrow b_f$
$\mathbf{E} \longleftarrow \mathsf{E}_0$

***Search for adjacent facets until the list L is empty.***

**while** $\mathbf{L} \neq \emptyset$ **do**
  Choose an element $((\mathsf{E}_r, a_r, b_r), (\mathsf{E}, a_f, b_f))$ from list $\mathbf{L}$
  $(\mathsf{E}^{\mathsf{adj}}, a_{\mathsf{adj}}, b_{\mathsf{adj}}) \longleftarrow ADJ((\mathsf{E}_r, a_r, b_r), (\mathsf{E}, a_f, b_f))$ Section 4
  $\mathbf{E_r} \longleftarrow RDG(\mathsf{E}^{\mathsf{adj}}, a_{\mathsf{adj}}, b_{\mathsf{adj}})$ Section 5
  **for each** element $(\mathsf{E_r}, a_r, b_r)$ in list $\mathbf{E_r}$ **do**
    **if** there exists an element $((\mathsf{A}_r, a_1, b_1), (\mathsf{A}, a_2, b_2))$ in list $\mathbf{L}$ such that $\mathsf{A}_r = \mathsf{E_r}$ **then**
      Remove element $((\mathsf{A}_r, a_1, b_1), (\mathsf{A}, a_2, b_2))$ from list $\mathbf{L}$
    **else**
      Add element $((\mathsf{E}_r, a_r, b_r), (\mathsf{E}^{\mathsf{adj}}, a_{\mathsf{adj}}, b_{\mathsf{adj}}))$ to list $\mathbf{L}$
    **end if**
  **end for**
  $G \longleftarrow \begin{bmatrix} G \\ a_{\mathsf{adj}}^T \end{bmatrix}$, $g \longleftarrow \begin{bmatrix} g \\ b_{\mathsf{adj}} \end{bmatrix}$
  $\mathbf{E} \longleftarrow \left(\mathbf{E}, \mathsf{E}^{\mathsf{adj}}\right)$
**end while**

***Report projection.***

Report $G$, $g$ and $\mathbf{E}$

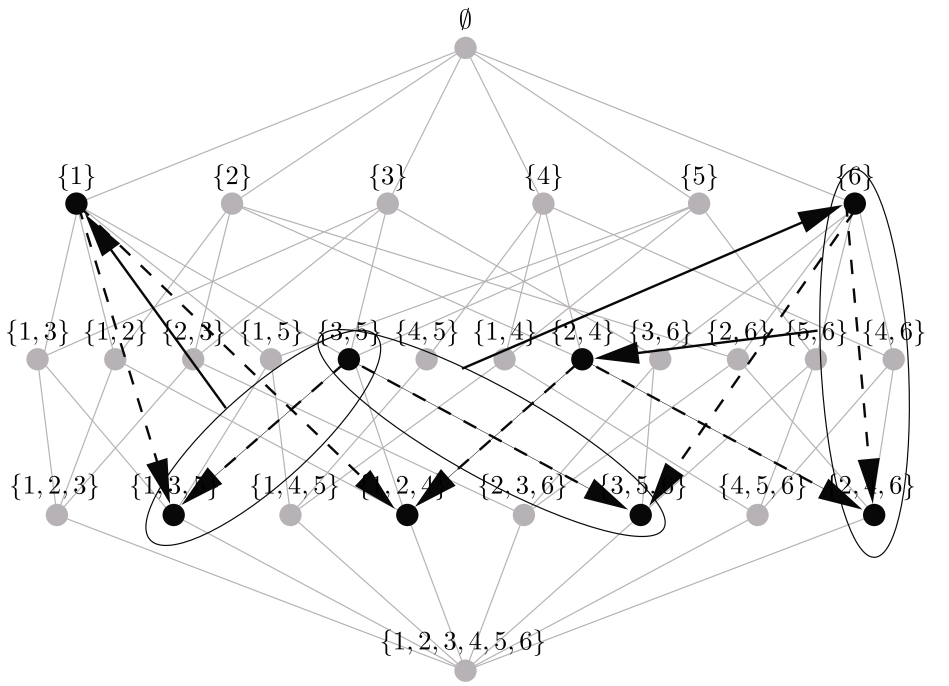



Figure 7: Example projection of a cube: Search path in cube face lattice

# 4 Adjacency Oracle

The goal of this section is to build the adjacency oracle $(\mathsf{E}^{\mathsf{adj}}, a_{\mathsf{adj}}, b_{\mathsf{adj}}) = ADJ((\mathsf{E}_r, a_r, b_r), (\mathsf{E}, a_f, b_f))$ introduced in Section 3. The oracle takes two equality sets $\mathsf{E}$ and $\mathsf{E_r} \supset \mathsf{E}$ that define a facet $\pi_d\,(P_{\mathsf{E}})$ and a ridge $\pi_d\,(P_{\mathsf{E_r}})$ of the projection $\pi_d\,(P)$. The unit normals $a_r$ and $a_f$ are orthogonal and have the property that:

$$\pi_d\,(P_{\mathsf{E}}) = \left\{x \;\middle|\; a_f^T x = b_f \right\} \cap \pi_d\,(P)\,,$$
$$\pi_d\,(P_{\mathsf{E_r}}) = \left\{x \;\middle|\; a_r^T x = b_r \right\} \cap \pi_d\,(P_{\mathsf{E}})\,.$$

The goal is to compute an equality set $\mathsf{E}^{\mathsf{adj}} \subset \mathsf{E}_r$ such that $\pi_d\,(P_{\mathsf{E}^{\mathsf{adj}}})$ is the facet of the projection that is adjacent to the given facet $\pi_d\,(P_{\mathsf{E}})$ and that contains the given ridge $\pi_d\,(P_{\mathsf{E_r}})$. Note that this facet is guaranteed to exist and to be unique by the Diamond Property (Theorem 7).

In this section we will show that the equality set of the adjacent facet is given by $\mathsf{E}^{\mathsf{adj}} \triangleq \{i \in \mathsf{E_r} \mid C_i x^\star + D_i y^\star = b_i\}$, where $(x^\star, y^\star)$ is the optimum of the LP

$$\begin{aligned}(x^\star, y^\star) = \quad \arg\max_{x,y} \quad & a_r^T x \\ \text{subject to} \quad & C_{\mathsf{E_r}} x + D_{\mathsf{E_r}} y \le b_{\mathsf{E_r}} \\ & a_f^T x = b_f(1-\epsilon),\end{aligned}$$

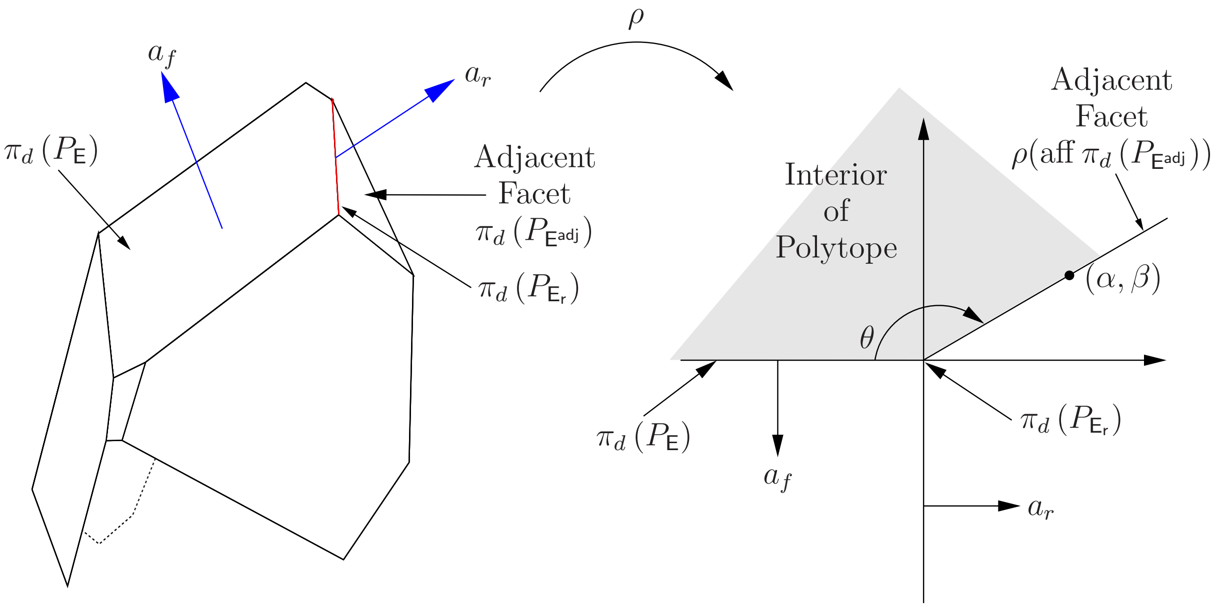



Figure 8: Map $\rho$

where $\epsilon$ is a small positive number. The remainder of this section is dedicated to proving this claim.

We will first show that as the adjacent facet must contain the given ridge, its affine hull can differ in only one degree of freedom from that of the given facet. We will introduce a linear mapping that will allow us to formulate the search over this degree of freedom as a linear program in order to determine a point on the affine hull of the adjacent facet. From this point we will then compute the equality set of the adjacent facet.

Similarly to [9], we define an affine transformation $\rho$ that maps $\mathbb{R}^d$ to $\mathbb{R}^2$:

$$\rho : \mathbb{R}^d \longrightarrow \mathbb{R}^2,\ x \longmapsto \begin{bmatrix} a_r^T \\ -a_f^T \end{bmatrix} (x - x_0), \tag{7}$$

where $x_0$ is any point in the affine hull of the given ridge aff $\pi_d(P_{\mathsf{E}_r})$.

The mapping $\rho$ takes every point in the polytope $\pi_d(P)$ to a two dimensional space such that all points in the affine hull of the facet $\pi_d(P_{\mathsf{E}})$ are mapped to the first axis and all points in the affine hull of the ridge aff $\pi_d(P_{\mathsf{E}_r})$ are mapped to the origin. The map is depicted in Figure 8 for a polytope that is projected to $\mathbb{R}^3$. For convenience, we will refer to the first axis as $\alpha$ and the second as $\beta$.

**Remark 23.** *It is assumed that the normal $a_f$ is given such that the ray $\lambda a_f$ will intersect the affine hull of $\pi_d(P_{\mathsf{E}})$ for a positive value of $\lambda$ and $a_r$ is outward facing from the facet $\pi_d(P_{\mathsf{E}})$. These requirements are satisfied if $a_f$ and $a_r$ are obtained from the ESP algorithm.*

*Under these assumptions, the facet $\pi_d(P_{\mathsf{E}})$ is mapped to the negative $\alpha$-axis and the interior of the polytope is strictly above the $\alpha$-axis as shown in Figure 8. Note that because the polytope $\pi_d(P)$ is convex, the angle that the adjacent facet makes with $\pi_d(P_{\mathsf{E}})$ must be less than $180^\circ$, or equivalently all points $(\alpha, \beta) \in \rho(\pi_d(P_{\mathsf{E}^{\mathsf{adj}}}))$ must have $\beta \geq 0$.*

We will now show that the affine hull of the adjacent facet $\operatorname{aff} \pi_d(P_{\mathsf{E}^{\mathsf{adj}}})$ will form a line going through the origin under the mapping $\rho$ and furthermore, all points in the polytope are mapped to the left of this line. First, the following lemma is needed.

**Lemma 24.** *If $R$ is a ridge, $F \supset R$ is a facet and* $\operatorname{aff} R = \left\{ x \,\middle|\, \begin{bmatrix} a_f & a_r \end{bmatrix}^T (x - x_0) = 0 \right\}$, *then there exists a $\gamma \in \mathbb{R}$ such that* $\operatorname{aff} F = \left\{ x \,\middle|\, \begin{bmatrix} 1 & \gamma \end{bmatrix} \begin{bmatrix} a_r^T \\ -a_f^T \end{bmatrix} (x - x_0) = 0 \right\}$.

*Proof.* $F$ is a facet and therefore only one equality is needed to describe its affine hull:

$$\operatorname{aff} F = \left\{ x \mid a^T (x - x_0) = 0 \right\}, \qquad \text{for some } a \in \mathbb{R}^d.$$

The ridge $R$ is a subset of $F$ and therefore for every $x$ in the affine hull of $R$, $x$ must be in the affine hull of $F$:

$$\begin{bmatrix} a_f & a_r \end{bmatrix}^T (x - x_0) = 0 \;\Rightarrow\; a^T (x - x_0) = 0.$$

This is equivalent to $x$ being in the affine hull of the facet if $x - x_0$ is in the left-nullspace of $\begin{bmatrix} a_f & a_r \end{bmatrix}$:

$$x - x_0 \in \mathsf{N}\left( \begin{bmatrix} a_f & a_r \end{bmatrix}^T \right) \;\Rightarrow\; a^T (x - x_0) = 0.$$

Finally, we can see that $a$ must be perpendicular to the left-nullspace of $\begin{bmatrix} a_f & a_r \end{bmatrix}$, or equivalently, in its rowspace:

$$a = a_r - \gamma a_f.$$

□

From Lemma 24 and (7) we can see that

$$\begin{aligned} \rho(\operatorname{aff} F) &= \left\{ \rho(x) \,\middle|\, \begin{bmatrix} 1 & \gamma \end{bmatrix} \rho(x) = 0 \right\} \\ &= \{ (\alpha, \beta) \mid \alpha + \gamma\beta = 0 \} \end{aligned}$$

and therefore under the mapping $\rho$, all points on the affine hull of the adjacent facet will be mapped to a line through the origin. The goal is now to determine which equality set $\mathsf{E}^{\mathsf{adj}} \subset \mathsf{E}_r$ defines this line.

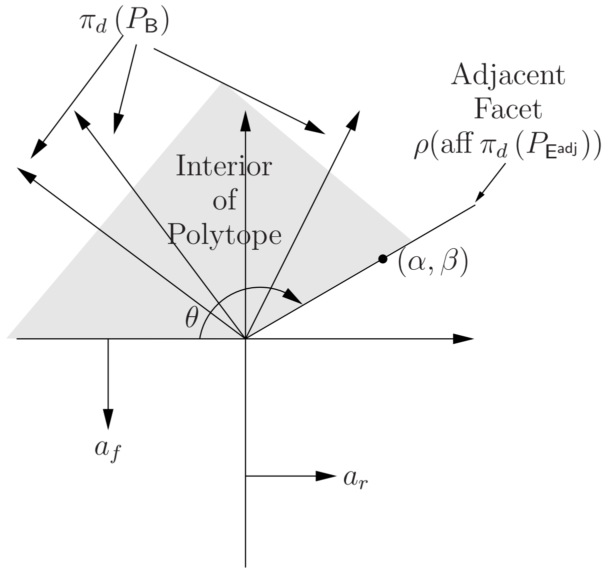



Figure 9: Subsets of $\mathsf{E}_r$ under the map $\rho$

Each subset $\mathsf{B}$ of $\mathsf{E_r}$ defines a polytope $\pi_d\left(P_{\mathsf{B}}\right)$ that, by Corollary 19, is a superset of the given ridge $\pi_d\left(P_{\mathsf{B}}\right) \supseteq \pi_d\left(P_{\mathsf{E_r}}\right)$ and correspondingly, the affine hull contains the affine hull of the given ridge $\operatorname{aff}\pi_d\left(P_{\mathsf{B}}\right) \supseteq \operatorname{aff}\pi_d\left(P_{\mathsf{E_r}}\right)$. Therefore, under $\rho$, the affine hull of $\pi_d\left(P_{\mathsf{B}}\right)$ will either map to the origin, or to a line through the origin. Each of these lines forms an angle with the negative $\alpha-$axis as shown in Figure 9. Since the portion of each of these lines that lies above the $\alpha-$axis must be internal to the projection of the polytope, the largest angle $\theta$ made with the negative $\alpha-$axis must define the adjacent facet.

**Remark 25.** *The simplest (and worst) approach to finding the adjacent facet would be to compute the projection $\pi_d\left(\operatorname{aff} P_{\mathsf{B}}\right)$ for each $\mathsf{B} \subset \mathsf{E}_r$ and then calculate the angle that it makes with $\pi_d\left(\operatorname{aff} P_{\mathsf{E}}\right)$. This would, of course, require a combinatorial number of calculations.*

**Remark 26.** *The adjacent facet can now be described in terms of the angle $\theta \in (0,\pi)$ between the adjacent facet and the given facet $\pi_d\left(P_{\mathsf{E}}\right)$ or by a point $(\alpha,\beta) \in \mathbb{R} \times \mathbb{R}_{>0}$ on the adjacent facet under the mapping $\rho$. Note that the affine mapping $\rho$ is conformal and therefore the angle $\theta$ defined under the mapping $\rho$ is the true angle between the two facets.*

The search for the largest angle is formulated as a maximisation over $\alpha$ while fixing $\beta$ to be a positive value; for convenience, we here choose $\beta = \epsilon b_f$, where $\epsilon$ is a small positive number. From Figure 10 this is clearly equivalent to maximizing $\theta$. In the following linear program we search for a point $(x, y)$ in the polytope $P$ and define the projection of this point (which is just $x$) under the mapping $\rho$ as $(\alpha,\beta) \triangleq \rho(x)$. We can then maximize $\alpha$ and constrain $\beta$ to

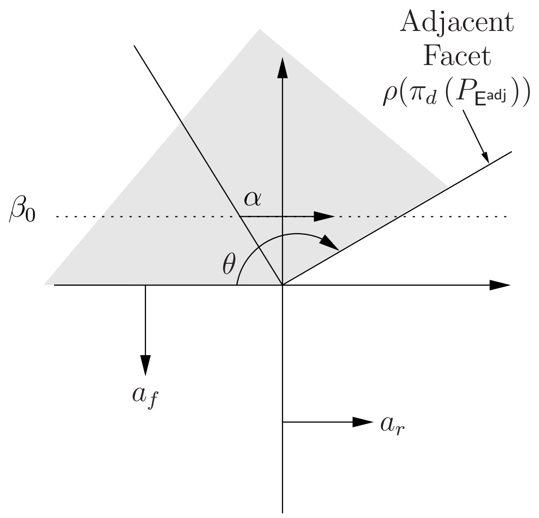



Figure 10: Linear optimisation under the wrapping map

be $\epsilon b_f$. The appropriate maximization is:

$$\begin{aligned} \max_{x,y} \quad & \alpha \\ \text{subject to} \quad & (x,y) \in P \\ & \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \rho(x) \\ & \beta = \epsilon b_f. \end{aligned} \tag{8}$$

From the definition of the mapping $\rho$ (7) and recalling that $a_f^T x_0 = b_f$ we can re-write LP (8) as follows:

$$\begin{aligned} J^\star = \quad \max_{x,y} \quad & a_r^T x \\ \text{subject to} \quad & C_{\mathsf{E_r}} x + D_{\mathsf{E_r}} y \leq b_{\mathsf{E_r}} \\ & a_f^T x = b_f(1-\epsilon). \end{aligned} \tag{9}$$

**Remark 27.** *Notice that in LP* (9) *only the constraints* $\mathsf{E_r}$ *are used to represent* $(x,y) \in P$. *From Corollary 19 we know that the adjacent facet must be a subset of* $\mathsf{E_r}$ *and therefore for efficiency we include only these constraints.*

We now show how to compute the equality set of the adjacent facet from the optimizer of LP (9). First, the following lemma is needed.

**Lemma 28.** *If* $(x^\star, y^\star)$ *is an optimizer of LP* (9) *and* $\mathsf{E}^\star = \{i \in \mathsf{E_r} \mid C_i x^\star + D_i y^\star = b_i\}$, *then* $\mathsf{E}^\star$ *is an equality set of* $P$, $\pi_d\,(P_{\mathsf{E}^\star}) \supset \pi_d\,(P_{\mathsf{E_r}})$ *and* $\dim \pi_d\,(P_{\mathsf{E}^\star}) = d-1$.

*Proof.* $\mathsf{E}^\star$ is clearly an equality set by Definition 8.

We first show that $\mathsf{E}^\star \subset \mathsf{E_r}$. Since LP (9) contains only the constraints $\mathsf{E_r}$, we have $\mathsf{E}^\star \subseteq \mathsf{E_r}$. To show proper inclusion we recall that the ridge $\pi_d(P_{\mathsf{E_r}})$ is mapped to the origin under the map $\rho$, however the constraint $\beta > 0$ implies that $(x^\star, y^\star)$ is not zero and therefore not on the affine hull of $P_{\mathsf{E_r}}$. It follows that $\mathsf{E_r} \neq \mathsf{E}^\star$.

From Theorem 13, $\mathsf{E}^\star \subset \mathsf{E_r}$ implies that $P_{\mathsf{E}^\star} \supset P_{\mathsf{E_r}}$ and therefore $\pi_d(P_{\mathsf{E}^\star}) \supseteq \pi_d(P_{\mathsf{E_r}})$. By construction, $x^\star \in \operatorname{aff} \pi_d(P_{\mathsf{E}^\star})$ and $x^\star \notin \pi_d(P_{\mathsf{E_r}})$ and therefore $\pi_d(P_{\mathsf{E}^\star}) \supset \pi_d(P_{\mathsf{E_r}})$.

It follows that $\dim \pi_d(P_{\mathsf{E}^\star}) > \pi_d(P_{\mathsf{E_r}}) = d-2$. $\dim \pi_d(P_{\mathsf{E}^\star})$ is $d$ if and only if $\mathsf{E}^\star = \emptyset$. From the constraints in LP (9), we can see that this is the case only if $x^\star = \infty$. However, the polytope $P$ is bounded and the angle between any two faces is less that $180°$ by convexity, and therefore $x^\star < \infty$ and $\pi_d(P_{\mathsf{E}^\star}) = d-1$. □

Recall that $P$ is bounded, and therefore the primal optimizer exists and is finite, although it may be non-unique. If the optimizer of LP (9) is unique, then we can compute the equality set of the adjacent facet directly using the following proposition. Note that a primal degenerate solution still provides a unique optimizer.

**Proposition 29.** *If $(x^\star, y^\star)$ is a unique optimal point of LP (9) and $\mathsf{E}^\star = \{i \in \mathsf{E_r} \mid C_i x^\star + D_i y^\star = b_i\}$, then $\mathsf{E}^\star$ is an equality set, $\dim P_{\mathsf{E}^\star} = d-1$ and $\pi_d(P_{\mathsf{E}^\star}) \supset \pi_d(P_{\mathsf{E_r}})$ is a facet of $\pi_d(P)$.*

*Proof.* By construction, the affine hull of $\pi_d(P_{\mathsf{E}^\star})$ is the affine hull of the adjacent facet. We now have that the adjacent facet is given by $F = (\operatorname{aff} \pi_d(P_{\mathsf{E}^\star})) \cap \pi_d(P)$. Clearly, if $F = \pi_d(P_{\mathsf{E}^\star})$, then $\pi_d(P_{\mathsf{E}^\star})$ is a facet.

By construction, the point $x^\star$ is in the interior of $F$. Since the optimal point is unique, there is only one $y = y^\star$ such that $(x^\star, y)$ is in the interior of $\pi_d^{-1}(F)$ and therefore the equality set of $\pi_d^{-1}(F)$ is given by $\mathsf{B} = \{i \mid C_i x^\star + D_i y^\star = b_i\} = \mathsf{E}^\star$. It follows that $F = \pi_d(P_{\mathsf{B}}) = \pi_d(P_{\mathsf{E}^\star})$ and $\pi_d(P_{\mathsf{E}^\star})$ is a facet of $\pi_d(P)$.

□

The solution is said to be dual-degenerate if there are multiple optimizers. In this situation Proposition 29 does not apply and more work is needed to compute the equality set of the adjacent facet. Recall that by construction, the projection of any point $(x^\star, y^\star)$ that is an optimizer of LP (9) is on the affine hull of the adjacent facet, $x^\star \in \operatorname{aff} \pi_d(P_{\mathsf{E}_{\mathsf{adj}}})$. Therefore, given any optimizer $(x^\star, y^\star)$ we can compute the affine hull of the adjacent facet $\operatorname{aff} \pi_d(P_{\mathsf{E}_{\mathsf{adj}}})$ from Lemma 24, which states that it has the form:

$$\begin{aligned}
\operatorname{aff} \pi_d(P_{\mathsf{E}_{\mathsf{adj}}}) &= \left\{x \;\middle|\; (a_f + \gamma a_r)^T (x - x_0) = 0\right\} \\
&= \left\{x \;\middle|\; (a_f + \gamma a_r)^T x = b_f + \gamma b_r\right\},
\end{aligned}$$

for some $\gamma \in \mathbb{R}$. The optimal point $x^\star$ is on the affine hull of the adjacent facet and therefore we can compute $\gamma$ from the above equation as:

$$\gamma = -\frac{a_r^T x^\star - b_r}{a_f^T x^\star - b_f}. \tag{10}$$

**Remark 30.** *Note that most commercial LP solvers will return an arbitrary optimizer* $(x^\star, y^\star)$ *in the case of dual-degeneracy.*

Finally, the equality set that defines the adjacent facet must be derived from the equation for its affine hull. The face $P_{\mathsf{E}^{\mathsf{adj}}}$ can be written as the pre-image of $\pi_d\left(P_{\mathsf{E}^{\mathsf{adj}}}\right)$:

$$P_{\mathsf{E}^{\mathsf{adj}}} = \pi_d^{-1}\left(\pi_d\left(P_{\mathsf{E}^{\mathsf{adj}}}\right)\right) = P \cap \left\{x \;\middle|\; (a_f + \gamma a_r)^T x = b_f + \gamma b_r)\right\}. \tag{11}$$

By definition the constraints of $P$ that are everywhere active in $P_{\mathsf{E}^{\mathsf{adj}}}$ form the equality set $\mathsf{E}^{\mathsf{adj}}$. A method for computing these constraints for a given polytope is given in Appendix A, which will provide the desired set, $\mathsf{E}^{\mathsf{adj}}$.

Algorithm 4.1 summarizes the procedure discussed in this section.

# 5 Ridge Oracle

The oracle $\mathsf{E_r} = RDG(\mathsf{E}, a_f, b_f)$ introduced in Section 3 will be developed here. The oracle takes as input an equality set $\mathsf{E}$ that defines a facet $\pi_d\left(P_{\mathsf{E}}\right)$ of the projection $\pi_d\left(P\right)$. The unit vector $a_f$ and the scalar $b_f$ also define the facet as $\pi_d\left(P_{\mathsf{E}}\right) = \left\{x \;\middle|\; a_f^T x = b_f\right\} \cap \pi_d\left(P\right)$. The oracle returns the equality sets of the ridges of the projection $\pi_d\left(P\right)$ that are subsets of the given facet $\pi_d\left(P_{\mathsf{E}}\right)$.

Recall that every face of a polytope is itself a polytope and therefore $\pi_d\left(P_{\mathsf{E}}\right)$ can be written in the form $\pi_d\left(P_{\mathsf{E}}\right) = \left\{x \;\middle|\; a_f^T x = b_f\right\} \cap \{x \mid \Upsilon x \le \upsilon\}$ for some $\Upsilon \in \mathbb{R}^{p \times d}$ and $\upsilon \in \mathbb{R}^p$. The ridges of the projection that are subsets of the given facet $\pi_d\left(P_{\mathsf{E}}\right)$ are then given by the facets of the polytope $\pi_d\left(P_{\mathsf{E}}\right)$, which can be written as $\pi_d\left(P_{\mathsf{E}}\right) \cap \{x \mid \Upsilon_i x = \upsilon_i\}$, for $i = 1, \ldots, p$. In this section we will show that if the dimension of the face $P_{\mathsf{E}}$ is $d-1$, then such an $\Upsilon$ and $\upsilon$ can be found directly, otherwise a low-dimensional projection is required.

We begin by computing expressions for the given face of the polytope $P_{\mathsf{E}}$ and its projection $\pi_d\left(P_{\mathsf{E}}\right)$. In Section 5.1, we show how to compute the equality sets of the ridges directly if the dimension of the face $P_{\mathsf{E}}$ is $d-1$ and then Section 5.2 handles the general case. The general case requires a recursive call to the ESP algorithm that reduces the dimension to which we are projecting at each step. Therefore in order to ensure that this recursion terminates, in Section 5.2.1 we present a method of computing the ridges directly when projecting to 1D.

**Lemma 31.** *If* $\mathsf{E}$ *is an equality set of* $P$ *such that* $\pi_d\left(P_{\mathsf{E}}\right)$ *is a facet of* $\pi_d\left(P\right)$ *and* $\dim P_{\mathsf{E}} =$

**Algorithm 4.1** Adjacency oracle

**Input:** Polytope $P$ and equality sets $\mathsf{E_r}$ and $\mathsf{E}$ such that $\pi_d(P_{\mathsf{E_r}})$ is a ridge and $\pi_d(P_{\mathsf{E}})$ is a facet of $\pi_d(P_{\mathsf{E}})$ and $\mathsf{E_r} \supset \mathsf{E}$.
Orthogonal unit vectors $a_f$ and $a_r$ and scalars $b_f$ and $b_r$ such that $\operatorname{aff} \pi_d(P_{\mathsf{E}}) = \left\{ x \;\middle|\; a_f^T x = b_f \right\}$ and $\operatorname{aff} \pi_d(P_{\mathsf{E_r}}) = \left\{ x \;\middle|\; a_r^T x = b_r \right\} \cap \operatorname{aff} \pi_d(P_{\mathsf{E}})$.

**Output:** Equality set $\mathsf{E^{adj}}$ such that $\pi_d(P_{\mathsf{E^{adj}}})$ is a facet of $\pi_d(P)$ and $\pi_d(P_{\mathsf{E_r}}) \subset \pi_d(P_{\mathsf{E^{adj}}})$

***Compute a point on the affine hull of the adjacent facet.***

Compute

$$\begin{aligned}(x^\star, y^\star) = \quad & \underset{x,y}{\arg\max} && a_r^T x \\ & \text{subject to} && C_{\mathsf{E_r}} x + D_{\mathsf{E_r}} y \le b_{\mathsf{E_r}} \\ & && a_f^T x = b_f(1-\epsilon)\end{aligned}$$

***Compute the equality set of the adjacent facet.***

**if** LP is *not* dual degenerate **then**
    $\mathsf{E^{adj}} \longleftarrow \{ i \in \mathsf{E_r} \mid C_i x^\star + D_i y^\star = b_i \}$
**else**
    $\gamma \longleftarrow -\frac{a_r^T x^\star - b_r}{a_f^T x^\star - b_f}$ (10)
    Compute the equality set $\mathsf{E^{adj}}$ of Appendix A

$$\left\{ (x,y) \mid C_{\mathsf{E_r}} x + D_{\mathsf{E_r}} y \le b_{\mathsf{E_r}},\ (a_f + \gamma a_r)^T x = b_f + \gamma b_r \right\}.$$

**end if**

***Compute affine hull of adjacent facet.***

$\begin{bmatrix} a_{\mathsf{adj}}^T & b_{\mathsf{adj}} \end{bmatrix} \longleftarrow \mathsf{N}\left(D_{\mathsf{E^{adj}}}{}^T\right)^T \begin{bmatrix} C_{\mathsf{E^{adj}}} & b_{\mathsf{E^{adj}}} \end{bmatrix}$ Proposition 21

***Normalize and ensure halfspace contains origin.***

$\begin{bmatrix} a_{\mathsf{adj}}^T & b_{\mathsf{adj}} \end{bmatrix} \longleftarrow \frac{\operatorname{sign} b_{\mathsf{adj}}}{\left\| a_{\mathsf{adj}} \right\|_2} \begin{bmatrix} a_{\mathsf{adj}}^T & b_{\mathsf{adj}} \end{bmatrix}$

***Report adjacent facet.***

Report $\left(\mathsf{E^{adj}}, a_{\mathsf{adj}}, b_{\mathsf{adj}}\right)$

*$d-1+n$, then the rank of $D_{\mathsf{E}}$ is $k-n$ and*

$$P_{\mathsf{E}} = \left\{ (x,y) \;\middle|\; \begin{array}{l} a_f^T x = b_f, \\ Sx + L\widetilde{y} \le t, \\ y = \mathsf{N}(D_{\mathsf{E}})\widetilde{y} + {D_{\mathsf{E}}}^{\dagger}(b_{\mathsf{E}} - C_{\mathsf{E}}x) \end{array} \right\},$$

*where* $\mathsf{E}^c \triangleq \{1,\ldots,q\} \setminus \mathsf{E}$*, and*

$$S \triangleq C_{\mathsf{E}^c} - D_{\mathsf{E}^c}{D_{\mathsf{E}}}^{\dagger}C_{\mathsf{E}}, \qquad L \triangleq D_{\mathsf{E}^c}\mathsf{N}(D_{\mathsf{E}}), \qquad t \triangleq b_{\mathsf{E}^c} - D_{\mathsf{E}^c}{D_{\mathsf{E}}}^{\dagger}b_{\mathsf{E}}.$$

*Proof.* The definition of $P_{\mathsf{E}}$ is

$$P_{\mathsf{E}} = \left\{ (x,y) \;\middle|\; \begin{array}{l} C_{\mathsf{E}}x + D_{\mathsf{E}}y = b_{\mathsf{E}}, \\ C_{\mathsf{E}^c}x + D_{\mathsf{E}^c}y \le b_{\mathsf{E}^c} \end{array} \right\}. \tag{12}$$

Let $\begin{bmatrix}\widehat{U} & \widetilde{U}\end{bmatrix}\begin{bmatrix}\Sigma & 0\\ 0 & 0\end{bmatrix}\begin{bmatrix}\widehat{V} & \widetilde{V}\end{bmatrix}^T \triangleq D_{\mathsf{E}}$ be the singular value decomposition of $D_{\mathsf{E}}$ and introduce the change of variables $\widehat{V}\widehat{y} + \widetilde{V}\widetilde{y} \triangleq y$. Multiplying by $\begin{bmatrix}\widehat{U} & \widetilde{U}\end{bmatrix}^T$, the affine hull of $P_{\mathsf{E}}$ can be written as the two equations:

$$\widehat{U}^T C_{\mathsf{E}}x + \Sigma\widehat{y} = \widehat{U}^T b_{\mathsf{E}} \tag{13}$$

$$\widetilde{U}^T C_{\mathsf{E}}x = \widetilde{U}^T b_{\mathsf{E}} \tag{14}$$

Recalling that $\widetilde{U}$ is equal to $\mathsf{N}\big({D_{\mathsf{E}}}^T\big)$ we see from Proposition 21 that (14) defines the affine hull of the projection: $a_f^T x = b_f$. Solving (13) for $\widehat{y}$ and substituting into (12) gives the desired result:

$$P_{\mathsf{E}} = \left\{ (x,y) \;\middle|\; \begin{array}{l} a_f^T x = b_f, \\ \big(C_{\mathsf{E}^c} - D_{\mathsf{E}^c}{D_{\mathsf{E}}}^{\dagger}C_{\mathsf{E}}\big)x + D_{\mathsf{E}^c}\widetilde{V}\widetilde{y} \le b_{\mathsf{E}^c} - D_{\mathsf{E}^c}{D_{\mathsf{E}}}^{\dagger}b_{\mathsf{E}}, \\ y = \widetilde{V}\widetilde{y} + {D_{\mathsf{E}}}^{\dagger}(b_{\mathsf{E}} - C_{\mathsf{E}}x) \end{array} \right\},$$

where we notice that the pseudo-inverse ${D_{\mathsf{E}}}^{\dagger}$ is given by $\widehat{V}\Sigma^{-1}\widehat{U}^T$.

It remains to be shown that the rank of $D_{\mathsf{E}}$ is $k-n$. Proposition 16 gives the dimension of the projection $\pi_d(P_{\mathsf{E}})$ as:

$$\dim \pi_d(P_{\mathsf{E}}) = \dim P_{\mathsf{E}} - k + \operatorname{rank} D_{\mathsf{E}}. \tag{15}$$

$\pi_d(P_{\mathsf{E}})$ is a facet and therefore the dimension of $\pi_d(P_{\mathsf{E}})$ is $d-1$ and the dimension of $P_{\mathsf{E}}$ has been assumed to be $d-1+n$, and therefore solving (15) provides the desired relation: $\operatorname{rank} D_{\mathsf{E}} = k-n$.

□

**Corollary 32.** *If* $\mathsf{E}$ *is an equality set of* $P$ *such that* $\pi_d(P_{\mathsf{E}})$ *is a facet of* $\pi_d(P)$ *and* $\dim P_{\mathsf{E}} = d-1+n$*, then*

$$\pi_d(P_{\mathsf{E}}) = \left\{x \;\middle|\; \exists \widetilde{y} \in \mathbb{R}^n,\ Sx + L\widetilde{y} \le t,\ a_f^T x = b_f \right\},$$

*where* $a_f, b_f, S, L,$*and* $t$ *are as defined in Lemma 31.*

*Proof.* Follows directly from Lemma 31. □

Note that if $n = 0$, Corollary 32 gives an explicit expression for $\pi_d(P_{\mathsf{E}})$ since $\widetilde{y} \in \mathbb{R}^0$. We now distinguish two cases and the following sections present methods for computing the ridges for $n = 0$ and $n > 0$ respectively.

**Remark 33.** *From Proposition 29, the dimension of the face* $P_{\mathsf{E}}$ *is* $d-1$ *if and only if the LP in* (9) *is not dual-degenerate.*

### 5.1 Case 1: $\dim P_{\mathsf{E}} = d-1$

In this section the assumption is made that the dimension of the face $P_{\mathsf{E}}$ is $d-1$. From Corollary 32 we can see that the facet is given by $\pi_d(P_{\mathsf{E}}) = \left\{x \;\middle|\; Sx \le t,\ a_f^T x = b_f \right\}$. We first prove that all facets of $\pi_d(P_{\mathsf{E}})$ (ridges of $\pi_d(P)$) can be written as $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ for some $i$ in $\mathsf{E}^c$. We then show how to select only those indices $i$ that correspond to facets of $\pi_d(P_{\mathsf{E}})$ (ridges of $\pi_d(P)$) and finally, how to construct the equality sets for each.

**Lemma 34.** *If* $\mathsf{E}$ *is an equality set of* $P$ *such that* $\pi_d(P_{\mathsf{E}})$ *is a facet of* $\pi_d(P)$ *and* $\dim P_{\mathsf{E}} = d-1$*, then for all* $i \in \mathsf{E}^c$*,*

$$\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right) = \left\{x \;\middle|\; \begin{bmatrix} a_f^T \\ S_i \end{bmatrix} x = \begin{bmatrix} b_f \\ t_i \end{bmatrix},\ Sx \le t \right\},$$

*where* $a_f$*,* $b_f$*,* $S$ *and* $t$ *are as defined in Lemma 31.*

*Proof.* The result follows directly from Corollary 32 by noting that if $n$ is zero then $\widetilde{y} \in \mathbb{R}^0$. □

From Corollary 32 and Lemma 34, we see that for each facet $F$ of $\pi_d(P_{\mathsf{E}})$, there exists an $i \in \mathsf{E}^c$ such that $F = \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$. It is clear from Lemma 34 and the definition of faces (Definition 5) that $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is a face of $\pi_d(P_{\mathsf{E}})$ for all $i \in \mathsf{E}^c$. Therefore, $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is a facet of $\pi_d(P_{\mathsf{E}})$ (a ridge of $\pi_d(P)$) if and only if it is of the appropriate dimension: $\dim \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right) = \dim \pi_d(P_{\mathsf{E}}) - 1 = d-2$.

The dimension of $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is $d-2$ if and only if the dimension of its affine hull is $d-2$, or equivalently the rank of $\begin{bmatrix} a_f^T & b_f \\ S_{\mathsf{Q}(i)} & t_{\mathsf{Q}(i)} \end{bmatrix}$ is two where $\mathsf{Q}(i)$ is the equality set of $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$. Therefore, we can see that $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is a ridge if and only if its equality set is given by:

$$\mathsf{Q}(i) \triangleq \left\{ j \in \mathsf{E}^c \;\middle|\; \operatorname{rank} \begin{bmatrix} a_f^T & b_f \\ S_{\{i,j\}} & t_{\{i,j\}} \end{bmatrix} = 2 \right\}. \tag{16}$$

This idea is formalized in Proposition 35 below.

**Proposition 35.** *If* $\mathsf{E}$ *is an equality set of* $P$ *such that* $\pi_d(P_\mathsf{E})$ *is a facet of* $\pi_d(P)$ *and* $\dim P_\mathsf{E} = d-1$*, then for all* $i \in \mathsf{E}^c$*,* $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ *is a facet of* $\pi_d(P_\mathsf{E})$ *if and only if* $\operatorname{rank}\begin{bmatrix} a_f^T & b_f \\ S_i & t_i \end{bmatrix} = 2$ *and there exists an* $x_0 \in \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ *such that* $S_j x_0 < t_j$ *for all* $j \in E^c \backslash \mathsf{Q}(i)$*, where*

$$\mathsf{Q}(i) \triangleq \left\{ j \in \mathsf{E}^c \;\middle|\; \operatorname{rank} \begin{bmatrix} a_f^T & b_f \\ S_{\{i,j\}} & t_{\{i,j\}} \end{bmatrix} = 2 \right\}.$$

*Proof.* Assume that $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is a facet of $\pi_d(P_\mathsf{E})$. We prove the existence of an $x_0$ such that $S_j x_0 < t_j$ for all $j \in \mathsf{E}^c \backslash \mathsf{Q}(i)$ by contradiction. Assume such an $x_0$ does not exist. Then there exists a $j \in \mathsf{E}^c \backslash \mathsf{Q}(i)$ such that $S_j x = t_j$ for all $x \in \pi_d(P_\mathsf{E})$. Since $j \notin \mathsf{E}^c \backslash \mathsf{Q}(i)$ we have that

$$\begin{aligned} \dim \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right) &= \dim \operatorname{aff} \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right) \\ &\leq \dim \left\{ x \;\middle|\; \begin{bmatrix} x_f^T \\ S_{\{i,j\}} \end{bmatrix} x = \begin{bmatrix} b_f \\ t_{\{i,j\}} \end{bmatrix} \right\} \\ &= d-3 \end{aligned}$$

and therefore an appropriate $x_0$ exists by contradiction.

Assume that there exists an $x_0$ such that $S_j x_0 < t_j$ for all $j \in \mathsf{E}^c \backslash \mathsf{Q}(i)$. We define the set

$$B(\epsilon, x_0) \triangleq \left\{ x \;\middle|\; \|x - x_0\|_2 < \epsilon, \begin{bmatrix} a_f^T \\ S_i \end{bmatrix} (x - x_0) = \begin{bmatrix} b_f \\ t_i \end{bmatrix} \right\}.$$

Note that $\dim B(\epsilon, x_0) = d-2$ for all $\epsilon > 0$ and all $x_0$. Clearly, there exists an $\epsilon > 0$ such that $(x_0 - x) \in \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ for all $x \in B(\epsilon, x_0)$. It follows that $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ has a subset of dimension $d-2$ and therefore $\dim \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right) \geq d-2$. The rank of $\begin{bmatrix} a_f^T & b_f \\ S_i & t_i \end{bmatrix}$ is two and therefore $\dim \pi_d\left(P_{\mathsf{E}\cup\{i\}}\right) \leq d-2$. It follows that $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is a $(d-2)$-face of $\pi_d(P_\mathsf{E})$ and is therefore a facet of $\pi_d(P_\mathsf{E})$.

□

We now propose a linear program that will test if $\mathsf{Q}(i)$ as defined in Proposition 35 is an equality set of $\pi_d(P)$:

$$\begin{aligned}
\underset{(\tau,x)}{\text{minimize}} \quad & \tau \\
\text{subject to} \quad & S_{\mathsf{Q}(i)} x \leq t_{\mathsf{Q}(i)} + \tau \\
& a_f^T x = b_f \\
& S_i x = t_i \\
& \tau \geq -c, \qquad \text{for some } c \in \mathbb{R}_{>0}
\end{aligned} \tag{17}$$

If there exists an $x$ and a strictly negative $\tau$ that satisfies LP (17), then clearly we have found an $x$ such that $S_j x < t_j$ for all $j \in \mathsf{E}^c \backslash Q(i)$ and the requirements of Proposition 35 are satisfied. Therefore $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is a facet of $\pi_d(P_{\mathsf{E}})$ if and only if there exists a feasible solution to LP (17) such that $\tau$ is strictly less than zero. Note that $x$ is bounded in LP (17) because the polytope $P$ is assumed bounded and the cost is bounded below. The equality set of the ridge $\pi_d\left(P_{\mathsf{E}\cup\{i\}}\right)$ is then given by the set $\mathsf{Q}(i)$.

## 5.2 Case 2: $\dim P_{\mathsf{E}} = d - 1 + n$, $n > 0$

Here we make the assumption that the dimension of the face $P_{\mathsf{E}}$ is larger than $d-1$. As a result, $n$ is strictly greater than zero and, from Corollary 32, we can write $\pi_d(P_{\mathsf{E}})$ as a projection from $\mathbb{R}^d \times \mathbb{R}^n$ to $\mathbb{R}^d$. Recall that each facet of $\pi_d(P_{\mathsf{E}})$ is a ridge of $\pi_d(P)$ and therefore computing this projection is equivalent to finding all the ridges that are subsets of $\pi_d(P_{\mathsf{E}})$. In this section we show how to compute this projection using ESP, which requires some slight work as it does not contain the origin and is not full-dimensional.

The polytope $\pi_d(P_{\mathsf{E}})$ can be written as the projection from $\mathbb{R}^d \times \mathbb{R}^n$ to $\mathbb{R}^d$:

$$\pi_d(P_{\mathsf{E}}) = \pi_d\left(\left\{(x,\widetilde{y}) \in \mathbb{R}^d \times \mathbb{R}^n \;\middle|\; Sx + L\widetilde{y} \leq t,\ a_f^T x = b_f\right\}\right), \tag{18}$$

where $S$, $L$ and $t$ are as defined in Corollary 32. Recall from Corollary 19 that every facet $F$ of $\pi_d(P_{\mathsf{E}})$ can be written as

$$\begin{aligned}
F &= (\pi_d(P_{\mathsf{E}}))_{\mathsf{R}} \\
&= \pi_d\left(\left\{(x,\widetilde{y}) \in \mathbb{R}^d \times \mathbb{R}^n \;\middle|\; S_{\mathsf{R}} x + L_{\mathsf{R}} \widetilde{y} = t_{\mathsf{R}},\ a_f^T x = b_f,\ Sx + L\widetilde{y} \leq t\right\}\right),
\end{aligned}$$

for some equality set $\mathsf{R}$ of $\pi_d(P_{\mathsf{E}})$. From Lemma 31 and Corollary 32 we can see that $\pi_d(P_{\mathsf{E}\cup\mathsf{R}}) = (\pi_d(P_{\mathsf{E}}))_{\mathsf{R}}$ and therefore $\mathsf{E}\cup\mathsf{R}$ is a ridge-defining equality set of $P$. All such facet-defining equality sets $\mathsf{R}$ of $\pi_d(P_{\mathsf{E}})$ can be computed via a call to the ESP algorithm, which therefore also gives all of the ridge-defining equality sets $\mathsf{E_r} = \mathsf{E}\cup\mathsf{R}$ of $P$ such that $\pi_d(P_{\mathsf{E_r}}) \subset \pi_d(P_{\mathsf{E}})$.

Note that (18) does not satisfy the assumptions that its projection is full dimensional or that the origin is contained in its interior, which is required if this projection is to be computed via ESP. Taking the singular value decomposition $a_f^T = \begin{bmatrix}\sigma & 0 & \cdots & 0\end{bmatrix} V^T$, we can re-write (18) as:

$$\begin{aligned}\pi_d\left(P_{\mathsf{E}}\right) &= \left\{ x \in \mathbb{R}^d \,\middle|\, \exists \widetilde{y} \in \mathbb{R}^n,\ \widetilde{x} \in \mathbb{R}^{d-1},\ Sx + L\widetilde{y} \leq t,\ x = V \begin{bmatrix} b_f/\sigma \\ \widetilde{x} \end{bmatrix} \right\} \\ &= \left\{ x \in \mathbb{R}^d \,\middle|\, x = V \begin{bmatrix} b_f/\sigma \\ \widetilde{x} \end{bmatrix},\ \widetilde{x} \in \pi_{d-1}\left(\widetilde{P}\right) \right\}, \end{aligned} \tag{19}$$

where

$$\widetilde{P} = \left\{ (\widetilde{x}, \widetilde{y}) \in \mathbb{R}^{d-1} \times \mathbb{R}^n \,\middle|\, S\widetilde{V}\widetilde{x} + L\widetilde{y} \leq t - S\widehat{V} b_f/\sigma \right\}. \tag{20}$$

and $V$ is partitioned appropriately as $V = \begin{bmatrix}\widehat{V} & \widetilde{V}\end{bmatrix}$.

Note that as the dimension of $\pi_d\left(P_{\mathsf{E}}\right)$ is $d-1$, $\widetilde{P}$ is guaranteed to be full-dimensional. The reader is referred to Appendix C for a simple procedure to ensure that the origin is interior to $\widetilde{P}$. The projection $\pi_{d-1}\left(\widetilde{P}\right)$ can now be computed using a recursive call to the ESP algorithm.

Finally, for each facet $\left\{ \widetilde{x} \,\middle|\, \widetilde{a}_f \widetilde{x} = \widetilde{b}_f \right\} \cap \pi_{d-1}\left(\widetilde{P}\right)$ of $\pi_{d-1}\left(\widetilde{P}\right)$, the corresponding ridge of $\pi_d\left(P\right)$ can be computed from (19) as:

$$\left\{ x \,\middle|\, \widetilde{a}_f \widetilde{V}^T x = \widetilde{b}_f \right\} \cap \pi_d\left(P_{\mathsf{E}}\right) \tag{21}$$

**Remark 36.** *The adjacency oracle requires that the equation describing each ridge has a unit normal and is orthogonal to the facet. Once the ridge* $\left\{ x \,\middle|\, a_r^T x = b_r \right\} \cap \pi_d\left(P_{\mathsf{E}}\right)$ *has been computed, an equivalent representation that has the required properties can be computed as:*

$$\begin{aligned}\begin{bmatrix} a_r^T & b_r \end{bmatrix} &\longleftarrow \begin{bmatrix} a_r^T & b_r \end{bmatrix} - a_f^T a_r \begin{bmatrix} a_f & b_f \end{bmatrix} \\ \begin{bmatrix} a_r^T & b_r \end{bmatrix} &\longleftarrow \frac{\operatorname{sign} b_r}{\left\| a_r \right\|_2} \begin{bmatrix} a_r^T & b_r \end{bmatrix}\end{aligned} \tag{22}$$

### 5.2.1 Projection to 1D

This section is included in order to guarantee that the recursion introduced when the dimension of $P_{\mathsf{E}}$ is larger than $d-1$ terminates.

If $d$ is one, then the facets of the projection are vertices and there are clearly only two of them. They can be computed by maximizing and minimizing along the $x-$axis as follows

$$(x^\star_{\min}, y^\star_{\min})/(x^\star_{\max}, y^\star_{\max}) \triangleq \begin{array}{ll} \min/\max\limits_{(x,y)} & x \\ \text{subject to} & Cx + Dy \leq b \end{array} \tag{23}$$

If LP (23) is not dual degenerate, then the equality sets of the ridges are given by

$$\begin{aligned} \mathsf{E}_{\min} &\triangleq \{i \mid Cx^\star_{\min} + Dy^\star_{\min} = b\}\,, \\ \mathsf{E}_{\max} &\triangleq \{i \mid Cx^\star_{\max} + Dy^\star_{\max} = b\}\,. \end{aligned}$$

If LP (23) is dual degenerate, then the equality sets must be computed as in Section 4. The pre-image of the facets $x^\star_{\min}$ and $x^\star_{\max}$ are given by

$$\begin{aligned} \pi_d^{-1}\left(x^\star_{\min}\right) = \pi_d^{-1}\left(P_{\mathsf{E}_{\min}}\right) = P \cap \{(x^\star_{\min}, y)\}\,, \\ \pi_d^{-1}\left(x^\star_{\max}\right) = \pi_d^{-1}\left(P_{\mathsf{E}_{\max}}\right) = P \cap \{(x^\star_{\max}, y)\}\,, \end{aligned}$$

where the equality sets $\mathsf{E}_{\min}$ and $\mathsf{E}_{\max}$ can be computed as in Appendix A.

Algorithm 5.1 describes the procedures developed in this section in algorithmic form.

# 6 Shooting Oracle

This section discusses the oracle $(\mathsf{E}_0, a_f, b_f) = SHOOT(P)$ introduced in Section 3. The goal of the oracle is to initialize the ESP algorithm by finding a random equality set $\mathsf{E}_0$ such that $\pi_d\left(P_{\mathsf{E}_0}\right)$ is a facet of the projection $\pi_d\left(P\right)$.

This oracle operates by randomly choosing a direction $\gamma \in \mathbb{R}^d$ and then solving an LP in order to move along the ray $\gamma r$, $r \geq 0$ until a point is found on the boundary of $\pi_d\left(P\right)$. If the point is not on a facet of $\pi_d\left(P\right)$ then the ray is randomly perturbed until it is. The active constraints that define the facet can then be calculated from the constraints of $P$ that are satisfied with equality at this point.

The following LP will be used to search in the direction of the ray $\gamma r$ until a facet of the projection is found:

$$(r^\star, y^\star) \triangleq \begin{array}{ll} \arg\max\limits_{(r,y)\in\mathbb{R}\times\mathbb{R}^k} & r \\ \text{subject to} & C\gamma r + Dy \leq b \end{array} \tag{24}$$

The point $(\gamma r^\star, y^\star)$ will be on the face of $P$ that projects to the face of $\pi_d\left(P\right)$ intersecting the ray $\gamma r$, where $r$ is a positive real number. This is equivalent to LP (9) where a point was computed on the adjacent facet. The method of computing the equality set of the face of $P$ that contains $(\gamma r^\star, y^\star)$ is the same as that presented in Section 4. Namely, if the optimizer

**Algorithm 5.1** Ridge oracle: $\mathbf{E_r} = RDG(\mathsf{E}, a_f, b_f)$

**Input:** Polytope $P$ and equality set $\mathsf{E}$ such that $\pi_d(P_{\mathsf{E}})$ is a facet of $\pi_d(P)$.

**Output:** List $\mathbf{E_r}$ whose elements are all equality sets $\mathsf{E}_r$ such that $\pi_d(P_{\mathsf{E}_r})$ is a facet of $\pi_d(P_{\mathsf{E}})$.

***Initialize variables.***

$\mathbf{E_r} \longleftarrow \emptyset$

Compute $S$, $L$ and $t$ as in Lemma 31

***Compute the dimension of $P_{\mathsf{E}}$.***

**if** rank $\begin{bmatrix} C_{\mathsf{E}} & D_{\mathsf{E}} \end{bmatrix} < k+1$ **then**

***Call ESP recursively to compute the ridges.***

    $\widetilde{\mathbf{E}}_{\mathbf{r}} \longleftarrow \mathrm{ESP}(\widetilde{P})$ (19)

    Convert the facets of $\pi_{d-1}\left(\widetilde{P}\right)$ into ridges of $\pi_d(P)$ and add to $\mathbf{E_r}$. (21)

**else**

***Test each $i \in \mathsf{E}^c$ to see if $\mathsf{E} \cup \{i\}$ defines a ridge.***

    **for each** $i$ in $\mathsf{E}^c$ **do**

        Compute $\tau^\star$ from LP (17)

        **if** $\tau^\star < 0$ **then** Proposition 35

            Compute equality set $\mathsf{Q}(i)$

            $\begin{bmatrix} a_r^T & b_r \end{bmatrix} \longleftarrow \begin{bmatrix} S_i & t_i \end{bmatrix}$

            $\mathbf{E_r} \longleftarrow ((\mathsf{Q}(i), a_r, b_r), \mathbf{E_r})$

        **end if**

    **end for**

**end if**

Normalize all equations and make orthogonal to the facet $\pi_d(P_{\mathsf{E}})$ Remark 36 of the LP is unique, then the equality set is given by $\mathsf{E}_0 \triangleq \{i \mid C_i\gamma r^\star + D_i y^\star = b_i\}$. However, if the LP is dual-degenerate, then the equality set can be computed by calculating the affine hull of the pre-image:

$$P_{\mathsf{E}_0} = \pi_d^{-1}\left(\pi_d\left(P_{\mathsf{E}_0}\right)\right) = P \cap \left\{x \;\middle|\; a_f^T x = b_f\right\},$$

where $a_f$ and $b_f$ are given by Proposition 21.

**Remark 37.** *The ESP algorithm assumes that the polytope $P$ contains the origin. This ensures that the ray $\gamma r$ will intersect the polytope for some value of $r \geq 0$. If an initial equality set is already known, then the interiority assumption can be relaxed.*

---

**Algorithm 6.1** Shooting oracle: Calculation of a random facet of $\pi_d(P)$

---

**Input:** Polytope $P = \left\{(x,y) \in \mathbb{R}^d \times \mathbb{R}^k \mid Cx + Dy \leq b\right\}$, which contains the origin in its interior and whose projection is full-dimensional.

**Output:** A randomly selected equality set $\mathsf{E}_0$ of $P$ such that $\pi_d\left(P_{\mathsf{E}_0}\right) \triangleq \left\{x \;\middle|\; a_f^T x = b_f\right\} \cap \pi_d(P)$ is a facet of $\pi_d(P)$.

***Find a face of $P$ that projects to a facet of $\pi_d(P)$***

1: **repeat**
2: Choose a random vector $\gamma \in \mathbb{R}^d$
3: Compute

$$(r^\star, y^\star) \triangleq \begin{array}{cl} \underset{(r,y)\in\mathbb{R}\times\mathbb{R}^k}{\arg\max} & r \\ \text{subject to} & C\gamma r + Dy \leq b \end{array}$$

4: $\mathsf{E}_0 \longleftarrow \{i \mid C_i\gamma r^\star + D_i y^\star = b_i\}$
5: **until** $\dim \pi_d\left(P_{\mathsf{E}_0}\right) = d - \operatorname{rank} \mathsf{N}\left({D_{\mathsf{E}_0}}^T\right)^T C_{\mathsf{E}_0} = d - 1$

***Compute affine hull of facet***

6: $\left[a_f^T \quad b_f\right] \longleftarrow \mathsf{N}\left({D_{\mathsf{E}_0}}^T\right)^T \left[C_{\mathsf{E}_0} \quad b_{\mathsf{E}_0}\right]$ — Proposition 21
7: $\left[a_f^T \quad b_f\right] \longleftarrow \frac{\operatorname{sign} b_f}{\left\|a_f\right\|_2}\left[a_f^T \quad b_f\right]$ — Remark 36

***Handle dual-degeneracy in LP*** — Section 4

8: **if** LP is dual-degenerate in 3 **then**
9: Compute equality set $\mathsf{E}_0$ such that $P_{\mathsf{E}_0} = \left\{(x,y) \;\middle|\; a_f^T x = b_f\right\} \cap P$ — Appendix A
10: **end if**

***Report facet***

11: Report $(\mathsf{E}_0, a_f, b_f)$.

---

## 7 Complexity Analysis

The complexity of the ESP algorithm is best described in terms of the number of linear programs needed. Let $LP(n,q)$ be the time complexity of an $LP$ of dimension $n$ with $q$ constraints. For the simplex approach, the average complexity is linear in $q$ and polynomial (approximately to the power three) in $n$, while the worst case is exponential. For interior point methods the worst case is polynomial to within a given $\epsilon$-bound of the optimum, however, there is a large constant multiplier on the interior point approach and as a result it can often be slower for 'small' problems (less than a few hundred dimensions).

First, assume that none of the linear programs computed during the adjacency oracle are dual-degenerate. If the polytope $P$ is in $\mathbb{R}^d \times \mathbb{R}^k$ and has $q$ constraints, then one LP in $d+k$ dimensions with $q$ constraints must be solved per output facet in order to compute its equality set. Computing the ridges of each facet requires a redundancy removal operation. This can be done in $q - |\mathsf{E_r}|$, $d$-dimensional LPs with $q - |\mathsf{E_r}|$ constraints. If $n_f$ is the number of inequalities in the projection $\pi_d(P)$, then the worst-case time complexity is

$$\mathcal{O}(n_f(LP(d+k,q) + (q - |\mathsf{E_r}|)LP(d, q - |\mathsf{E_r}|))).$$

If the polytope is in general position, then $|\mathsf{E_r}|$ is $k$ for all facets and the complexity becomes:

$$\mathcal{O}(n_f(LP(d+k,q) + (q-k)LP(d,q-k))). \tag{25}$$

Clearly, the complexity is linear in the number of output facets.

It is interesting to note that if the dimension that we are projecting to and the number of constraints are held constant, then the complexity becomes $\mathcal{O}(n_f LP(k,q))$. Finally, if the size of the polytope $P$ is fixed, then the complexity is linear is the number of output facets, $\mathcal{O}(n_f)$.

If degeneracy is encountered, then ESP may need to be called recursively. The number of calls can be significant depending on the polytope and for certain polytopes, other projection algorithms may well be more efficient.

## 8 Numerical Simulations

This section shows comparative simulations of ESP against other commonly used projection algorithms. Each algorithm is presented with an irredundant halfspace description of a polytope $P \triangleq \left\{(x,y) \in \mathbb{R}^d \times \mathbb{R}^k \mid Cx + Dy \le b\right\}$ and the output of the algorithm is an irredundant halfspace description of the projection $\pi_d(P) = \left\{x \in \mathbb{R}^d \;\middle|\; \exists y \in \mathbb{R}^k \; (x,y) \in P\right\}$.

Four algorithms have been compared:

- Fourier Elimination: This algorithm was implemented in C as described in [15].

- Projection Cone: If the projection cone is defined as $W \triangleq \{v \mid vD = 0, v \geq 0\}$, then the projection is given by $\pi_d(P) = \{x \mid (vC)x \leq vb,\ \forall v \in \operatorname{extr} W\}$. The extreme rays of $W$ were computed via the double description method CDD, which is implemented in C [8].

- Vertex Enumeration: In this approach, the vertices of the polytope $P$ are enumerated, projected and then the convex hull of the projection is calculated. All computations are done via the double description method CDD [8].

- ESP: The ESP algorithm is implemented in Matlab® and all linear programs are computed via the Stanford Systems Optimization Laboratory (SOL) toolbox [12].

As we require the output to be irredundant, each inequality generated by the first two approaches is tested for redundancy by a call to a linear program. The LP code used is the Stanford Systems Optimization Laboratory (SOL) toolbox [12].

The polytopes that are used in this comparison are randomly generated, bounded by a hypersphere of radius 10 and all inequalities in their description are irredundant. Figure 11 shows the results of the simulations where the $y-$axis is a logarithmic scale of the time taken per facet of the projection and the $x-$axis is the variable $\xi$, which is defined in the following list. Let $q$ be the number of halfspaces in the description of $P$ and we define four scenarios:

a. $\mathbb{R}^{10} \longrightarrow \mathbb{R}^4, \quad q = \xi, \quad \xi = 10, \ldots, 100$
   This test demonstrates the complexity of the algorithms as a function of the number of halfspaces in the polytope. From (25) and recalling that an LP has linear complexity for fixed dimension, the complexity per facet of the projection of ESP is $\mathcal{O}(q^2)$; this tendency can be seen in Figure 11(a).

b. $\mathbb{R}^{\xi} \longrightarrow \mathbb{R}^4, \quad q = 3\xi, \quad \xi = 5, \ldots, 30$
   In this test we project to a fixed dimension while increasing the dimension of the polytope $P$. From (25), the complexity of ESP should increase as $\mathcal{O}(LP(k, 3k))$. It is clear from Figure 11(b) that for these polytopes, ESP is capable of projecting from much higher dimensions than the other approaches. The main limitation as the dimension increases is the rapid growth of the number of facets in the projection.

c. $\mathbb{R}^{\xi} \longrightarrow \mathbb{R}^4, \quad q = 2\xi, \quad \xi = 5, \ldots, 70 \qquad P$ is a hypercube
   This scenario computes the projection of rotated hypercubes to $\mathbb{R}^4$. Note that while a $n-$dimensional hypercube has $2n$ facets, it contains $2^n$ vertices. Therefore, the projection cone and vertex enumeration approaches are ill-suited in this case. However, as can be seen from Figure 11(c), ESP can handle very high dimensions.

d. $\mathbb{R}^{\xi} \longrightarrow \mathbb{R}^2, \quad q = 3\xi, \quad \xi = 5, \ldots, 50$
   The final test shows the projection of randomly generated polytopes from high dimensions to $\mathbb{R}^2$. The procedure described in Section 5.2.1 for the calculation of ridges in 1D makes ESP particularly suited to this task, as can be readily seen from Figure 11(d).

**Remark 38.** *Note that difference in speed between ESP and the other algorithms for small problems is likely to be due in large part to the overhead involved in the implementation of ESP in Matlab®.*

All code presented in this report can be found online at
`http://www-control.eng.cam.ac.uk/~cnj22/projection.html`

# 9 Conclusion

This paper has presented the ESP algorithm, which is a new approach for the projection of polytopes in halfspace form. The proposed algorithm requires a linear number of linear programs per output facet in order to compute the projection. If the size of the polytope is kept constant, it has been shown that the complexity becomes linear in the number of facets of the projection.

The algorithm has been implemented and some comparative simulations were given in Section 8. These simulations use a sample of a few classes of polytopes and we do not claim that ESP is the fastest approach in every case. For example, there is a large class of polytopes for which the projection cone has a very small number of extreme rays or the number of vertices of the polytope is small, such as in a simplex. However, the simulations presented here demonstrate that there is a large class of polytopes for which ESP is particularly suited and is able to calculate projections of polytopes in significantly higher dimensions than existing methods. Of particular note are high dimensional polytopes that are to be projected to a low dimension, polytopes represented by a small number of inequalities and hypercubes.

# 10 Acknowledgements

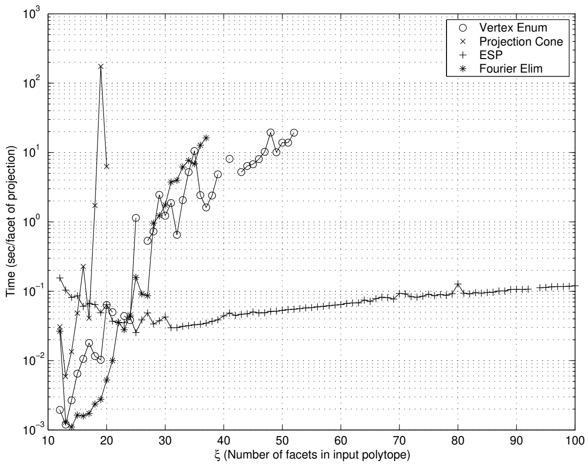
The authors would like to thank Pascal Greider for his valuable comments early in this work.

Colin Jones would like to thank the Natural Sciences and Engineering Research Council of Canada and the Princes Commonwealth Trust for their support during this work.

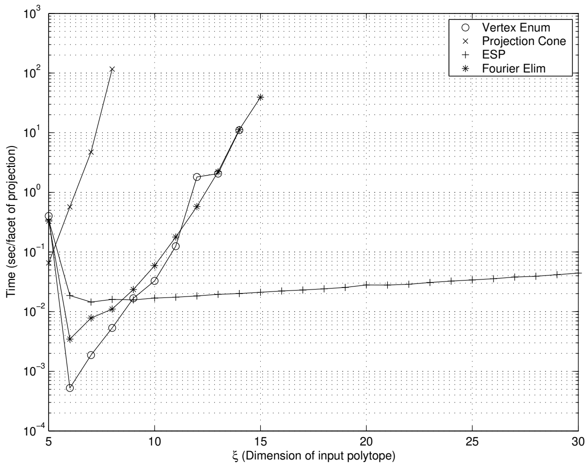
Eric Kerrigan is a Royal Academy of Engineering Post-doctoral Research Fellow and would like to thank the Royal Academy of Engineering for supporting this research.

# A Calculation of the Affine Hull

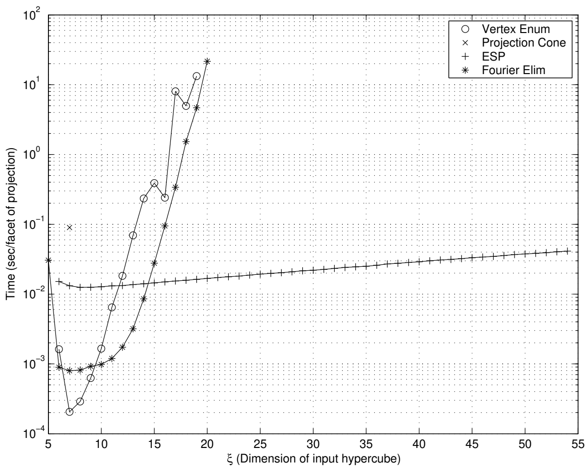
This section presents a well known algorithm for the computation of the equality set $\mathsf{E}$ of a polytope $P$ such that $P_{\mathsf{E}} = P$. The input to the algorithm is the matrix $A \in \mathbb{R}^{q \times n}$ and the vector $b \in \mathbb{R}^q$ that defines the polytope $P \triangleq \{z \in \mathbb{R}^n \mid Az \leq b\}$. The output is an equality set $\mathsf{E}$ such that $P_{\mathsf{E}} = P$ and the affine hull is given by aff $P = \{z \mid A_{\mathsf{E}} z = b_{\mathsf{E}}\}$.



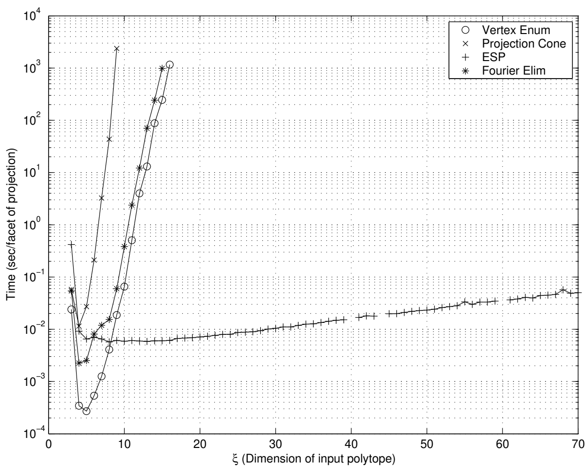

(a) $\mathbb{R}^{10} \longrightarrow \mathbb{R}^4$; $\xi$ = number of facets in $P$




(b) $\mathbb{R}^{\xi} \longrightarrow \mathbb{R}^4$; $3\xi$ = number of facets in $P$




(c) $\mathbb{R}^{\xi} \longrightarrow \mathbb{R}^4$; $P$ is a rotated hypercube




(d) $\mathbb{R}^{\xi} \longrightarrow \mathbb{R}^2$; $3\xi$ = number of facets in $P$

Figure 11: Comparative Simulation Results for Randomly Generated Polytopes

Recall that the equality set of $P$ consists of all constraints that are active at every point in the polytope. Therefore, if a point can be found in the polytope for which a given constraint is not active, then that constraint is clearly not in the equality set. Given a constraint $i \in \{1, \ldots, q\}$, a point exists for which it is not active if $A_i z - b_i$ is strictly less than zero for some $z$. We can search for such a point by minimizing the value of $A_i z - b_i$:

$$\begin{aligned} J(i)^\star \triangleq \quad & \min_z && A_i z - b_i \\ & \text{subject to} && Az \leq b. \end{aligned}$$

If $J(i)^\star$ is strictly negative, then constraint $i$ is not in the equality set. If it is positive, then the constraint is redundant and if it is equal to zero then it is in the equality set.

# B Projection of non Full-Dimensional Polytopes

If $\pi_d(P)$ is not full dimensional, then $P$ must have a non-trivial affine hull and there exists an equality set $\mathsf{A}$ such that $P_\mathsf{A} = P$. An algorithm for computing the affine hull of a polytope, and the equality set of the constraints that determine it is given in Appendix A. If $\mathsf{A}$ is the equality set of the affine hull, then from Proposition 21 we can write the projection $\pi_d(P)$ as

$$\pi_d(P) = \{x \in \mathbb{R}^n \mid \exists y,\ Fx = f,\ Cx + Dy \leq b\}, \tag{26}$$

where $F \triangleq \mathsf{N}\big({D_\mathsf{A}}^T\big)^T C_\mathsf{A}$ and $f \triangleq \mathsf{N}\big({D_\mathsf{A}}^T\big)^T b_\mathsf{A}$. We now form a polytope $\widetilde{P}$ that satisfies the assumption that its projection is full-dimensional and from which we can recover the projection $\pi_d(P)$.

The equation $Fx = f$ can be equivalently written as $x = \mathsf{N}(F)\widetilde{x} + F^\dagger f$, for $\widetilde{x} \in \mathbb{R}^{\dim \pi_d(P)}$, where we note that $\dim \pi_d(P) = n - \operatorname{rank} F$. The polytope $\pi_d(P)$ then becomes

$$\begin{aligned} \pi_d(P) &= \left\{ x \in \mathbb{R}^n \,\middle|\, \exists y,\ x = \mathsf{N}(F)\widetilde{x} + F^\dagger f,\ (\widetilde{x}, y) \in \widetilde{P} \right\} \\ &= \left\{ x \in \mathbb{R}^n \,\middle|\, x = \mathsf{N}(F)\widetilde{x} + F^\dagger f,\ \widetilde{x} \in \pi_{\dim \pi_d(P)}\left(\widetilde{P}\right) \right\} \end{aligned}$$

where

$$\widetilde{P} \triangleq \left\{ (\widetilde{x}, y) \,\middle|\, C\mathsf{N}(F)\widetilde{x} + Dy \leq b - CF^\dagger f \right\}.$$

We now show that $\widetilde{P}$ satisfies the assumption that $\pi_{\dim \pi_d(P)}\left(\widetilde{P}\right)$ is full-dimensional.

$$\begin{aligned}
&\pi_{\dim \pi_d(P)}\left(\operatorname{aff} \widetilde{P}\right) \\
&\quad = \pi_{\dim \pi_d(P)}\left(\left\{\widetilde{x} \;\middle|\; \exists y,\ C_{\mathsf{A}}\mathsf{N}(F)\widetilde{x} + D_{\mathsf{A}}y = b_{\mathsf{A}} - C_{\mathsf{A}}F^{\dagger}f\right\}\right) \\
&\quad = \left\{\widetilde{x} \;\middle|\; \mathsf{N}\left(D_{\mathsf{A}}{}^{T}\right)^{T} C_{\mathsf{A}}\mathsf{N}(F)\widetilde{x} = \mathsf{N}\left(D_{\mathsf{A}}{}^{T}\right)^{T}\left(b_{\mathsf{A}} - C_{\mathsf{A}}F^{\dagger}f\right)\right\} \\
&\quad = \mathbb{R}^{\dim \pi_d(P)} \qquad (27)
\end{aligned}$$

where (27) follows because $F$ is defined as $\mathsf{N}\left(D_{\mathsf{A}}{}^{T}\right)^{T} C_{\mathsf{A}}$.

Once the projection of $\widetilde{P}$ is computed using ESP, an expression is needed to recover $\pi_d(P)$. $\widetilde{x}$ can be written as a function of $x$ as $\mathsf{N}(F)^{T}x = \widetilde{x}$. If $\pi_{\dim \pi_d(P)}\left(\widetilde{P}\right)$ is given by $\{\widetilde{x} \mid G\widetilde{x} \le g\}$ then

$$\begin{aligned}
\pi_d(P) &= \left\{x \;\middle|\; x = \mathsf{N}(F)\widetilde{x} + F^{\dagger}f,\ G\widetilde{x} \le g\right\} \\
&= \left\{x \;\middle|\; Fx = f,\ G\mathsf{N}(F)^{T}x \le g\right\}.
\end{aligned}$$

# C Projection of Polytopes that do not Contain the Origin

In this section we consider a polytope $P$ that does not contain the origin in its interior. The solution presented here is to compute a strictly interior point $(x_0, y_0)$ and then translate the polytope such that this point gets mapped to the origin. After running ESP on the translated polytope, the projection is then translated back by $-x_0$ to restore the origin placement.

The following linear program will compute an interior point of polytope $P$ whose equality set is $\mathsf{A}$:

$$\begin{aligned}
(x_0, y_0) \triangleq\ & \underset{(\tau,x,y)}{\operatorname{argmin}} && \tau \\
& \text{subject to} && C_{\mathsf{A}^c}x + D_{\mathsf{A}^c}y \le b_{\mathsf{A}^c} + \tau \\
& && C_{\mathsf{A}}x + D_{\mathsf{A}}y = b_{\mathsf{A}}
\end{aligned} \qquad (28)$$

Translating $P$ by $(x_0, y_0)$ gives $\widehat{P} \triangleq \{(x, y) \mid Cx + Dy \le b + Cx_0 + Dy_0\}$. If the projection of $\widehat{P}$ is $\pi_d\left(\widehat{P}\right) = \left\{x \;\middle|\; \widehat{G}x \le \widehat{g}\right\}$, then $\pi_d(P) = \left\{x \;\middle|\; \widehat{G}x \le \widehat{g} - \widehat{G}x_0\right\}$.

## References


[1] Nina Amenta and Günter M. Ziegler. Shadows and slices of polytopes. In *Proceedings of the twelfth annual symposium on computational geometry*, pages 10–19. ACM Press, 1996.

[2] D. Avis. lrs: A revised implementation of the reverse search vertex enumeration algorithm. In G. Kalai and G.M. Ziegler, editors, *Polytopes — Combinatorics and Computation*, pages 177–198. Birkhauser-Verlag, 2000.

[3] Egon Balas. Projection with a minimum system of inequalities. *Computational Optimization and Applications*, 10:189–193, 1998.

[4] Egon Balas and Maarten Oosten. On the dimension of projected polyhedra. *Discrete Applied Mathematics*, 87:1–9, 1998.

[5] Michel L. Balinski. On the graph structure of convex polyhedra in n-space. *Pacific Journal of Math*, 95(11):431–434, 1961.

[6] F. Blanchini. Set invariance in control. *Automatica*, 35(11):1747–1767, 1999. Survey Paper.

[7] Herbert Edelsbrunner and Ernst Peter Mücke. Simulation of simplicity: A technique to cope with degenerate cases in geometric algorithms. *ACM Transactions on Graphics*, 9(1):66–104, 1990.

[8] K. Fukuda and A. Prodon. Double description method revisited. In M. Deza, R. Euler, and I. Manoussakis, editors, *Combinatorics and Computer Science*, volume 1120 of *Lecture Notes in Computer Science*, pages 91–111. Springer-Verlag, 1996. Postscript file available from ftp://ftp.ifor.math.ethz.ch/pub/fukuda/reports/ddrev960315.ps.gz.

[9] Komei Fukuda, Thomas M. Liebling, and Christine Lütolf. Extended convex hull. In $12^{th}$ *Canadian Conference on Computational Geometry*, pages 57–64, July 2000.

[10] Joxan Jaffar, Michael J. Maher, Peter J. Stuckey, and Roland H. C. Yap. Projecting clp($\mathcal{R}$) constraints. *New Generation Computing*, 11(3,4):449–469, 1993.

[11] Eric Kerrigan. *Robust Constraint Satisfaction: Invariant Sets and Predictive Control*. PhD thesis, University of Cambridge, 2000.

[12] Walter Murray and Michael Saunders. Systems Optimization Laboratory (SOL). http://www.sbsi-sol-optimize.com.

[13] Jean Ponce, Steve Sullivan, Attawith Sudsang, Jean-Daniel Boissonnat, and Jean-Pierre Merlet. On computing four-finger equilibrium and force-closure grasps of polyhedral objects. *International Journal of Robotics Research*, February 1995.

[14] R. Tyrrell Rockafellar. *Convex Analysis*. Princeton University Press, 1970.

[15] Vijay Saraswat and Pascal Van Hentenryck, editors. *Principles and Practice of Constraint Programming*, chapter 14, pages 245–268. The MIT Press, 1995.

[16] S.N. Černikov. Contraction of finite systems of linear inequalities (in russian). *Doklady Akademiia Nauk SSSR*, 152(5):1075–1078, 1963. (English translation in *Societ Mathematics Doklady,* Vol. 4, No. 5 (1963), pp.1520-1524).

[17] R. Vidal, S. Schaffert, J. Lygeros, and S. Sastry. *Hybrid Systems: Computation and Control*, volume 1790 of *Lecture Notes in Computer Science*, chapter Controlled Invariance of Discrete Time Systems, pages 437–450. Springer Verlag, 2000.

[18] Roger Webster. *Convexity*. Oxford University Press, 1994.

[19] Günter M. Ziegler. *Lectures on Polytopes.* Springer-Verlag, New York, first edition, 1995.