



Improved strategies of the Equality Set Projection (ESP) algorithm for computing polytope projection

Binbin Pei^{1,3}, Wenfeng Xu², and Yinghui Li³

¹School of Electronic Engineering, Xidian University, Xi'an 710071, China

²National Key Laboratory of Electromagnetic Energy, Wuhan 430033, China

³Aeronautics Engineering College, Air Force Engineering University, Xi'an 710038, China

Correspondence: Wenfeng Xu (xwfd96@163.com)

Received: 28 February 2023 – Revised: 5 September 2023 – Accepted: 2 November 2023 – Published: 19 March 2024

Abstract. This paper proposes an optimization method for the Equality Set Projection algorithm to compute the orthogonal projection of polytopes. However, its computational burden significantly increases for the case of dual degeneracy, which limits the application of the algorithm. Two improvements have been proposed to solve this problem for the Equality Set Projection algorithm: first, a new criterion that does not require a discussion of the uniqueness of the solution in linear programming, which simplifies the algorithm process and reduces the computational cost; and second, an improved method that abandons the calculation of a ridge's equality set to reduce the computational burden in the case of high-dimensional dual degeneracy.

1 Introduction

Orthogonal projection is valuable for both theoretical research (Kwan et al., 2022; Xu and Guo, 2021) and engineering applications (Xia et al., 2021; Liu et al., 2023), particularly as a basic operator for polytopes, which are a type of geometric structure composed of flat boundaries and giving a good representation of linear constraints on variables. The orthogonal projection of polytopes is extensively applied in the field of control and optimization (Borrelli et al., 2017; Lee and Kouvaritakis, 2000; Sadeghzadeh, 2018). For example, the calculations of the Minkowski sum (Teissandier and Delos, 2011), controllable set (Vincent and Wu, 1990), reachable set (Rakovic et al., 2006), and parametric linear program (Jones et al., 2008) can all be transformed into the projection of polytopes.

Traditional projection algorithms can be grouped into three classes, i.e., Fourier elimination (Keerthi and Sridharan, 1990; Saraswat and Hentenryck, 1995), block elimination (Balas, 1998; Fukuda and Prodon, 1996), and vertex enumeration (Avis, 1998). The former two algorithms require expensive computation and produce a lot of redundant inequalities. The third algorithm is suitable for polytopes with fewer vertices; however, the number of vertices

of a high-dimensional polytope exceeds the number of half-spaces. Thus, it is not an efficient algorithm for the high-dimensional cases.

The Equality Set Projection (ESP) algorithm was proposed in Jones et al. (2004), and it applies the inspiration of the gift-wrapping algorithm to the projection calculation. Under the condition of non-degeneracy, the complexity of ESP is linear in the number of half-spaces, which offers a tremendous advantage compared to the traditional projection algorithms. However, in the case of dual degeneracy, the computation grows exponentially.

Dual degeneracy is a frequently encountered phenomenon in control applications. For instance, when variables have constant maximum and minimum constraints, the occurrence of dual degeneracy is more likely. Unfortunately, this limitation curtails the wider application of ESP algorithms in the control field. Therefore, the primary motivation behind enhancing the ESP method is to improve its performance under dual-degeneracy conditions. We propose a novel criterion based on the non-zero rows of the null-space matrix to address the issue of discussing the uniqueness of solutions under such conditions. Furthermore, we introduce an improved strategy that eliminates the need to search for equation sets under dual-degeneracy conditions. These two advancements

render the ESP algorithm more concise, efficient, and easily implementable. Consequently, the algorithm's performance in the situations involving dual degeneracy has been significantly enhanced, which paves the way for further promotion and application of ESP in engineering practices.

The paper is organized as follows. The preliminary knowledge and the workflow of the original ESP algorithm are introduced in Sect. 2. The two proposed improvement strategies are given in Sects. 3 and 4. Numerical simulation comparisons between the original ESP and the improved ESP are reported in Sect. 5, and final conclusions given in Sect. 6.

2 Preliminaries and existing results

2.1 Definitions and notation

1. A polytope is a bounded polyhedron defined by the intersection of closed half-spaces. Let P equal $\{(x, y) \in R^d \times R^k | Cx + Dy \leq b\}$; the orthogonal projection of P onto x space is defined as

$$\pi_d(P) = \left\{ x \in R^d | \exists y \in R^k, (x, y) \in P \right\}.$$

2. f is a face of the polytope $P \subset R^d \times R^k$ if there exists a hyperplane $\{(x, y) | c^T x + d^T y = b\}$, where $c \in R^d$, $d \in R^k$, and $b \in R$ such that

$$f = P \cap \left\{ (x, y) \in R^d \times R^k | c^T x + d^T y = b \right\}$$

and $c^T x + d^T y \leq b$ for all $(x, y) \in P$. The face of the polytope $P \subset R^d \times R^k$ refers to the face of dimension $n - 1$, whereas the ridge refers to the face of dimension $n - 2$.

3. The affine hull of every face of the polytope is

$$\text{aff } f = \left\{ (x, y) \in R^d \times R^k | c^T x + d^T y = b \right\}.$$

4. If A is a matrix and $E \subseteq \{1, \dots, q\}$, then A_E is a matrix formed by rows of A whose indices are in E . If P is a polytope, then the notation P_E refers to the set $P_E = P \cap \{z | C_E z + D_E y = b_E\}$.

5. E is an equality set of P if and only if $E = G(E)$, where

$$G(E) = \{i \in \{1, \dots, q\} | C_i x + D_i y = b_i, \forall (x, y) \in P_E\}.$$

6. $N(A)$ is defined as a matrix whose columns form an orthonormal basis for the null space of A .

2.2 Auxiliary results

The following results can be found in Jones (2005), Ziegler (1995), Webster (1994), and Balas and Oosten (1998), respectively. Their proofs are omitted for the sake of brevity.

Theorem 2.1 (Jones, 2005) Given a polytope

$$P = \left\{ (x, y) \in R^d \times R^k | Cx + Dy \leq b \right\},$$

if E is an equality set of P , then P_E is a face of P . Furthermore, if F is a face of P , then there exists a unique equality set such that $F = P_E$.

Proposition 2.1 (Ziegler, 1995) If $P \subset R^d \times R^k$ is a polytope, then the projection of P onto R^d is a polytope.

Proposition 2.2 (Webster, 1994) If F_1, \dots, F_t are facets of a polytope $P = \{z \in R^n | Az \leq b\}$, $\text{aff } F_i = \{z \in R^n | a_i^T z = b_i\}$, where each $a_i \in R^n$, $b_i \in R$, then

$$P = \text{aff } P \cap \left\{ x \in R^n \mid \begin{bmatrix} a_1^T \\ \vdots \\ a_t^T \end{bmatrix} x \leq \begin{bmatrix} b_1 \\ \vdots \\ b_t \end{bmatrix} \right\}.$$

Proposition 2.3 (Ziegler, 1995) If $P \subset R^d \times R^k$ is a polytope, then for every face F of $\pi_d(P)$, the pre-image $\pi_d^{-1}(F)$ is a face of P .

Proposition 2.4 (Jones, 2005) If $M = \{(x, y) \in R^d \times R^k | C_E x + D_E y = b_E\}$ is a affine set, then the projection of M onto R^d is

$$\pi_d(M) = \left\{ x \in R^d \mid N(D_E^T)^T C_E x = N(D_E^T)^T b_E \right\}.$$

Proposition 2.5 (Balas and Oosten, 1998) If E is an equality set of polytope

$$P = \left\{ (x, y) \in R^d \times R^k | Cx + Dy \leq b \right\},$$

then $\dim \pi_d(P_E) = \dim P_E - k + \text{rank } D_E$.

2.3 Original ESP algorithm

We introduce the workflow of the original ESP algorithm in this subsection. A brief introduction is given here, and the details can be found in Jones (2005).

The algorithm is initialized by discovering a random facet F_π in $\pi_d(P)$. Then we compute all ridges contained in the initial facet and search all facets adjacent to it through each ridge. Subsequently, we compute all the ridges of each new facet and let the iterating process continue until all the ridges appear twice. According to the property of the polytope, each ridge exactly connects to two facets. Therefore, all facets of $\pi_d(P)$ can be found by this iteration. From Proposition 2.2, we finally get $\pi_d(P) = \{x \in R^d | Gx \leq g\}$.

The ESP algorithm consists of three key oracles as given below. Shooting Oracle:

$$(E_{f_0}, a_f, b_f) = \text{SHOOT}(P).$$

Ridge Oracle:

$$L E_r = \text{RDG}(E_f, a_f, b_f).$$

Adjacency Oracle:

$$(E_{\text{adj}}, \mathbf{a}_{\text{adj}}, b_{\text{adj}}) = \text{ADJ}((E_f, \mathbf{a}_f, b_f)(E_r, \mathbf{a}_r, b_r)).$$

The SHOOT oracle randomly obtains an equality set E_{f_0} of P satisfying $\pi_d(P_{E_{f_0}}) = \{\mathbf{x} | \mathbf{a}_f^T \mathbf{x} = b_f\} \cap \pi_d(P)$ as a facet of $\pi_d(P)$, the RDG oracle returns all ridges contained in the facet (E_f, \mathbf{a}_f, b_f) of $\pi_d(P)$, and the ADJ oracle takes two equality sets E_r and E_f of P and returns the unique equality set E_{adj} , where $\pi_d(P_{E_{\text{adj}}}) = \{\mathbf{x} | \mathbf{a}_f^T \mathbf{x} = b_f\} \cap \pi_d(P)$ is a facet of $\pi_d(P)$ and $\pi_d(P_{E_r}) = \{\mathbf{x} | \mathbf{a}_r^T \mathbf{x} = b_r\} \cap \pi_d(P_{E_{\text{adj}}})$ is a ridge of $\pi_d(P)$ with $\pi_d(P_{E_r}) \subset \pi_d(P_{E_{\text{adj}}})$.

2.3.1 Outline of the ESP algorithm

Step 1. Get an initial facet $(E_{f_0}, \mathbf{a}_f, b_f)$ of P : $(E_{f_0}, \mathbf{a}_f, b_f) = \text{SHOOT}(P)$. Then, all the ridges contained in the facet are calculated: $LE_r = \text{RDG}(E_{f_0}, \mathbf{a}_f, b_f)$. For each element $(E_{r_j}, \mathbf{a}_{r_j}, b_{r_j})$ in LE_r , record $((E_{f_0}, \mathbf{a}_{f_0}, b_{f_0}), (E_{r_j}, \mathbf{a}_{r_j}, b_{r_j}))$ in list L and let $\mathbf{G} = \mathbf{a}_f^T$, $g = b_f$.

Step 2. If $L \neq \emptyset$, select an element $((E_{f_i}, \mathbf{a}_{f_i}, b_{f_i}), (E_{r_i}, \mathbf{a}_{r_i}, b_{r_i}))$ randomly from list L and let $((E_{f_n}, \mathbf{a}_{f_n}, b_{f_n}), (E_{r_n}, \mathbf{a}_{r_n}, b_{r_n})) = ((E_{f_i}, \mathbf{a}_{f_i}, b_{f_i}), (E_{r_i}, \mathbf{a}_{r_i}, b_{r_i}))$. If $L = \emptyset$, stop and report \mathbf{G} and g .

Step 3. Search the adjacent facet such that $(E_{\text{adj}}, \mathbf{a}_{\text{adj}}, b_{\text{adj}}) = \text{ADJ}((E_{f_n}, \mathbf{a}_{f_n}, b_{f_n}), (E_{r_n}, \mathbf{a}_{r_n}, b_{r_n}))$ and let

$$\mathbf{G} = \begin{bmatrix} \mathbf{G} \\ \mathbf{a}_{\text{adj}}^T \end{bmatrix}, \quad \mathbf{a} = \begin{bmatrix} \mathbf{a} \\ b_{\text{adj}} \end{bmatrix}.$$

Then, all the ridges contained in the adjacent facet can be calculated: $LE_r = \text{RDG}(E_{\text{adj}}, \mathbf{a}_{\text{adj}}, b_{\text{adj}})$. For each element $(E_{r_j}, \mathbf{a}_{r_j}, b_{r_j})$ in LE_r , if there exists an element in L that already contains E_{r_j} , we remove the element from the list. Otherwise, we add element $((E_{r_j}, \mathbf{a}_{r_j}, b_{r_j}), (E_{\text{adj}}, \mathbf{a}_{\text{adj}}, b_{\text{adj}}))$ to list L . Go to Step 2.

2.3.2 SHOOT oracle

Step 1 (selecting a direction $\boldsymbol{\gamma}$ and finding a point on the surface of P)

Randomly choose $\boldsymbol{\gamma} \in R^d$ and solve the following linear programming.

$$\begin{aligned} (r^*, \mathbf{y}^*) &= \arg \max_{(r, \mathbf{y}) \in R \times R^k} r, \\ \text{s.t. } \mathbf{C}\boldsymbol{\gamma}r + \mathbf{D}\mathbf{y} &\leq \mathbf{b} \end{aligned} \quad (1)$$

Step 2 (determining whether a facet of $\pi_d(P)$ is found)

Get E_{f_0} as $E_{f_0} = \{i | \mathbf{C}_i \boldsymbol{\gamma}r + \mathbf{D}_i \mathbf{y} = b_i\}$. If $\text{Rank}(\mathbf{N}(\mathbf{D}_{E_{f_0}}^T)^T \mathbf{C}_{E_{f_0}}) = 1$, go to Step 3. Otherwise, go to Step 1.

Step 3 (computing the affine hull of the facet and normalizing)

$$\begin{aligned} [\mathbf{a}_f^T, b_f] &= \mathbf{N}(\mathbf{D}_{E_{f_0}}^T)^T [\mathbf{C}_{E_{f_0}}, b_{E_{f_0}}] \\ [\mathbf{a}_f^T, b_f] &= \frac{\text{sign } b_f}{\|\mathbf{a}_f\|_2} [\mathbf{a}_f^T, b_f] \end{aligned}$$

Step 4 (handling the dual degeneracy in Step 1)

If the linear programming (1) in Step 1 is not dual-degenerate, i.e., (1) has a unique optimizer, then stop. Otherwise, E_{f_0} needs to be recalculated as follows.

For each row $\mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{y} \leq b_i$ of P , we compute the following linear programming problem:

$$J^*(i) = \min_{\mathbf{x}, \mathbf{y}} \mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{y} - b_i,$$

$$\text{s.t. } \mathbf{C}_{E_c} \mathbf{x} + \mathbf{D}_{E_c} \mathbf{y} \leq \mathbf{b},$$

$$\mathbf{x} = \mathbf{x}^*,$$

where

$$E_c^i = \{1, 2, \dots, q\} \setminus i, \quad E_{f_0} = \{i \in \{1, 2, \dots, q\} | J^*(i) = 0\}.$$

2.3.3 RDG oracle

Step 1 (computing the dimension of P_{E_f})

If $k + d - \text{rank}[\mathbf{C}_{E_f} \ \mathbf{D}_{E_f}] = d - 1$, continue. Otherwise, go to step 3.

Step 2 ($\dim P_{E_f} = d - 1$) For all $i \in E_f^c = \{1, 2, \dots, q\} \setminus E_f$,

$$Q_i = \left\{ j \in E^c \mid \text{rank} \begin{bmatrix} \mathbf{a}_f^T & b_f \\ \mathbf{S}_{\{i, j\}} & t_{\{i, j\}} \end{bmatrix} = 2 \right\},$$

and, solving the following linear programming,

$$\begin{aligned} \min_{(\tau, \mathbf{x})} & \tau(i), \\ \text{s.t. } & \begin{cases} \mathbf{S}_{Q_c}(\mathbf{x}) \leq t_{Q_c}(\mathbf{x}) + \tau, \\ \mathbf{a}_f^T \mathbf{x} = b_f, \\ \mathbf{S}_i \mathbf{x} = t_i, \\ \tau(i) \geq -c, \end{cases} \end{aligned}$$

where c is an arbitrary positive constant, $Q_c(i) = E_c \setminus Q(i)$. If $\tau(i) < 0$, then $\mathbf{a}_r = \mathbf{S}_{\{i\}}$, $b_r = t_{\{i\}}$ and add $(Q(i), \mathbf{a}_r, b_r)$ to LE_r .

Step 3 ($\dim P_{E_f} > d - 1$)

Conduct singular value decomposition $\mathbf{a}_f^T = [\sigma \ 0 \ \dots \ 0] \mathbf{V}^T$, $\mathbf{V} = [\hat{\mathbf{V}} \ \tilde{\mathbf{V}}]$. Then, we get

$$\tilde{P} = \{(\tilde{\mathbf{x}}, \tilde{\mathbf{y}}) \in R^{d-1} \times R^n | \mathbf{S}_f \tilde{\mathbf{V}} \tilde{\mathbf{x}} + \mathbf{L}_f \tilde{\mathbf{y}} \leq t_f - \mathbf{S} \hat{\mathbf{V}} b_f / \sigma\},$$

where $\mathbf{S}_f = \mathbf{C}_{E_f^c} - \mathbf{D}_{E_f^c} \mathbf{D}_{E_f}^\dagger \mathbf{C}_{E_f}$, $\mathbf{L}_f = \mathbf{D}_{E_f^c} \mathbf{N}(\mathbf{D}_E)$, and $t_f = b_{E_f^c} - \mathbf{D}_{E_f^c} \mathbf{D}_{E_f}^\dagger b_{E_f}$. The $\pi_{d-1}(\tilde{P})$ can be computed using a recursive call to the ESP algorithm. Each facet of $\pi_{d-1}(\tilde{P})$ can be converted into the ridge of $\pi_d(P)$ as

$$\{\mathbf{x} | \tilde{\mathbf{V}} \tilde{\mathbf{a}}_f^T \mathbf{x} = \tilde{b}_f\} \cap \pi_{d-1}(\tilde{P}).$$

Then, we get $\mathbf{a}_r = \tilde{\mathbf{a}}_f, b_r = \tilde{b}_f$, and $(Q(i), \mathbf{a}_r, b_r)$ to LE_r .

Step 4 (normalizing and orthogonalizing)

$$\begin{aligned} \begin{bmatrix} \mathbf{a}_r^T & b_r \end{bmatrix} &\leftarrow \begin{bmatrix} \mathbf{a}_r^T & b_r \end{bmatrix} - \mathbf{a}_f^T \mathbf{a}_r \begin{bmatrix} \mathbf{a}_f^T & b_f \end{bmatrix} \\ \begin{bmatrix} \mathbf{a}_r^T & b_r \end{bmatrix} &\leftarrow \frac{\text{sign } b_r}{\|\mathbf{a}_r\|_2} \begin{bmatrix} \mathbf{a}_r^T & b_r \end{bmatrix} \end{aligned}$$

2.3.4 ADJ oracle

Step 1 (finding a point on the adjacent facet $\pi_d(P_{E_{\text{adj}}})$ and its corresponding pre-image)

Solve the following linear programming:

$$\begin{aligned} (\mathbf{x}^*, \mathbf{y}^*) &= \arg \max_{\mathbf{x}, \mathbf{y}} \mathbf{a}_r^T \mathbf{x}, \\ \text{s.t. } \mathbf{C}_{E_r} \mathbf{x} + \mathbf{D}_{E_r} \mathbf{y} &\leq b_{E_r}, \\ \mathbf{a}_f^T \mathbf{x} &= b_f (1 - \varepsilon), \end{aligned} \tag{2}$$

where ε is a sufficiently small positive constant.

If LP (2) is not dual-degenerate, i.e., LP (2) has a unique optimizer $(\mathbf{x}^*, \mathbf{y}^*)$, then continue. Otherwise, go to Step 3.

Step 2 (the case without dual degeneracy)

$$E_{\text{adj}} \leftarrow \{i \in E_r \mid \mathbf{C}_i \mathbf{x}^* + \mathbf{D}_i \mathbf{y}^* = b_i\}$$

Go to Step 4.

Step 3 (the case of dual degeneracy)

$$\gamma = -\frac{\mathbf{a}_r^T \mathbf{x}^* - b_r}{\mathbf{a}_f^T \mathbf{x}^* - b_f}$$

For each row $\mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{y} \leq b_i$ of P , compute the following linear programming problem:

$$J^*(i) = \min_{\mathbf{x}, \mathbf{y}} \mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{y} - b_i,$$

$$\text{s.t. } \mathbf{C}_{E_c} \mathbf{x} + \mathbf{D}_{E_c} \mathbf{y} \leq b_{E_c},$$

$$(\mathbf{a}_r + \gamma \mathbf{a}_f)^T \mathbf{x} = b_r + \gamma b_f,$$

where $E_c^i = \{1, 2, \dots, q\} \setminus \{i\}, E_{\text{adj}} = \{i \mid J^*(i) = 0\}$.

Step 4 (computing the affine hull of the facet and normalizing)

$$\begin{aligned} \begin{bmatrix} \mathbf{a}_f^T, b_f \end{bmatrix} &= \mathbf{N}(\mathbf{D}_{E_0}^T)^T \begin{bmatrix} \mathbf{C}_{E_0}, b_{E_0} \end{bmatrix} \\ \begin{bmatrix} \mathbf{a}_f^T, b_f \end{bmatrix} &= \frac{\text{sign } b_f}{\|\mathbf{a}_f\|_2} \begin{bmatrix} \mathbf{a}_f^T, b_f \end{bmatrix} \end{aligned}$$

3 Improvement of the uniqueness discussion

In the original ESP algorithm, expressions (1) and (2) in the SHOOT and ADJ oracles need to judge the uniqueness of the optimizer of linear programming (LP), and different steps are taken according to whether the optimizer is unique or not.

In this section, an illustrative example is given at first, the barriers of the original ESP algorithm in the case of dual degeneracy are listed, and the alternative criteria are proposed.

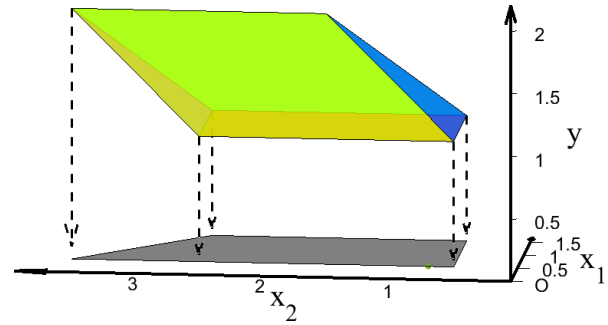


Figure 1. Polytope and its projection onto R^d .

3.1 An illustrative example

The linear programming (2) in Step 1 of the ADJ oracle is given for illustration, and the solution of (2) satisfies

$$\mathbf{x}^* \in \pi_d(E_{\text{adj}}) \cap \left\{ \mathbf{x} \mid \mathbf{a}_f^T \mathbf{x} = b_f (1 - \varepsilon) \right\}. \tag{3}$$

The projection of $(\mathbf{x}^*, \mathbf{y}^*)$ is on $\pi_d(P_{E_{\text{adj}}})$ (Jones, 2005). If the optimizer $(\mathbf{x}^*, \mathbf{y}^*)$ of linear programming is unique, E_{adj} should be taken as the equality set corresponding to the face $P_{E_{\text{adj}}}$ where $(\mathbf{x}^*, \mathbf{y}^*)$ is located. If the optimizer is not unique, E_{adj} should be taken as the equality set corresponding to the face $P_{E_{\text{adj}}}$, where the entire optimal solution space is located. Next, a simple example is provided to illustrate the reasons for the discussion of the uniqueness of the optimizer and leads to its alternative strategy.

Example 1

Given a polytope $P = \{(\mathbf{x}, \mathbf{y}) \in R^d \times R^k \mid \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{y} \leq \mathbf{b}\}$,

$$\mathbf{C} = \begin{bmatrix} 0 & -2 \\ 0 & 0.667 \\ 0.4 & 0 \\ -2 & 0 \\ 0 & 0 \end{bmatrix}, \mathbf{D} = \begin{bmatrix} 2 \\ -0.667 \\ 0.4 \\ 0 \\ -1 \end{bmatrix}, \mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix},$$

the orthogonal projection $\pi_d(P) = \{\mathbf{G}\mathbf{x} \leq \mathbf{g}\}$ onto \mathbf{x} space R^d is shown in Fig. 1, where

$$\mathbf{G} = \begin{bmatrix} -1 & 0 \\ 0.707 & 0.707 \\ 1 & 0 \\ 0 & -1 \end{bmatrix}, \mathbf{g} = \begin{bmatrix} -0.5 \\ 2.83 \\ 1.5 \\ -0.5 \end{bmatrix}.$$

Figure 2 shows that the ADJ oracle takes the facet $\{\mathbf{a}_f^T \mathbf{x} = b_f\} \cap \pi(P)$ of P (the blue line segment in this example) and ridge $\{\mathbf{a}_r^T \mathbf{x} = b_r\} \cap \pi_d(P_{E_f})$ (the intersection of the blue line segment and the yellow line segment in this example) to solve the adjacent facet (the yellow line segment in this example is the facet to be solved), and \mathbf{x}^* is the intersection of plane $\mathbf{a}_f^T \mathbf{x} = b_f (1 - \varepsilon)$ and the yellow line segment.

All the points on the red line segment of P can be projected onto \mathbf{x}^* , which indicates that all points on the red line

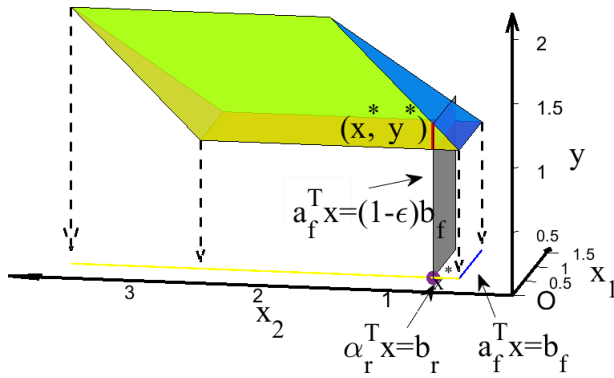


Figure 2. The ADJ oracle in Example 1.

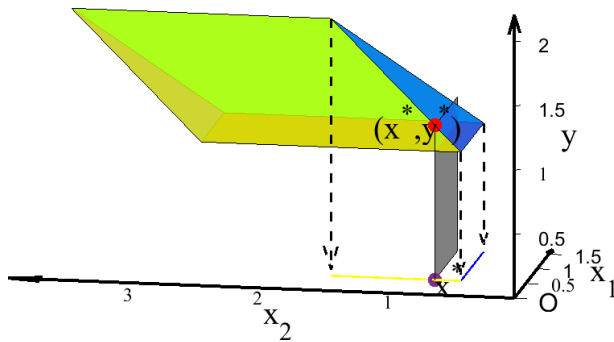


Figure 3. The non-uniqueness of the optimal solution is ignored.

segment are optimal solutions of the LP problem. The optimal solution domain belongs to the yellow facet of $P_{E_{adj}}$, and its corresponding equality set is $E_{adj} = \{4\}$.

However, most of the LP solvers will only give one optimal solution for the LP problem with multiple optimal solutions. If this solution is mistakenly considered the only optimal solution, this may lead to the iteration cycle and increase the computational burden.

In this case, if (x^*, y^*) on the intersection of the blue facet and yellow facet is taken as the unique optimal solution, the $P_{E'_{adj}}$ that (x^*, y^*) belongs to is the intersection line of the blue and yellow facets of P , $E'_{adj} = \{1, 4\}$. According to Fig. 3, $P_{E'_{adj}} \subset P_{E_{adj}}$ and $\pi_d(P_{E'_{adj}}) \subset \pi_d(P_{E_{adj}})$.

Owing to $\text{aff}\pi_d(P_{E'_{adj}}) = \text{aff}\pi_d(P_{E_{adj}})$, some facets may appear multiple times in the following iterative process, which may increase the number of iterations, and even the algorithm cannot be terminated.

3.2 Shortcomings of the original algorithm

The barriers of the original ESP algorithm in the case of dual degeneracy can be listed as follows.

- i. Currently, most of the LP solvers do not have the ability to judge the uniqueness of the solutions of linear pro-

gramming. Although several achievements have been made on the unique conditions of LP solutions (Mangasarian, 1979), these methods are difficult to embed into the existing LP solvers, since they induce additional computation.

- ii. Since the dual degeneracy is only a necessary but insufficient condition for linear programming to have multiple optimizers (Borrelli et al., 2017), it is conservative to determine the uniqueness of the solution by testing whether the dual degeneracy occurs in Jones et al. (2004).
- iii. Expression (3) implies that, in the case of $d \geq 3$, the value of x^* is always not unique; i.e., the LP problem always has multiple optimal solutions. However, if only the value of x^* is not unique and the value of y^* is unique, irrespective of the point in the optimal solution space, it can be regarded as the unique optimal solution, and the obtained $P_{E_{adj}}$ does not vary. In this case, an extra computation is added to handle the non-uniqueness of the optimal solution of the LP.

3.3 Alternative criteria

According to Example (1), there may be multiple faces P_{E_f} in P , and the projections of their affine hulls are all $\text{aff}F_\pi$. The collection of all E_f satisfying this condition can be denoted as

$$\begin{aligned} \Xi_{F_\pi} &= \left\{ E_f \subset \{1, 2, \dots, q\} \mid \pi_d(\text{aff}P_{E_f}) \right. \\ &= \left. \text{aff}F_\pi, G(E_f) = E_f \right\}, \end{aligned} \quad (4)$$

where

$$\begin{aligned} G(E_f) &= \{i \in \{1, 2, \dots, q\} \mid C_i x + D_i y = b_i, \\ &\forall (x, y) \in P_{E_f}\}. \end{aligned}$$

The discussion on the uniqueness of optimizers of LP is to ensure that the obtained E_f is the element in Ξ_{F_π} with the maximal $\dim P_{E_f}$. The primary result can be stated as given below.

Theorem 3.1 Consider the Ξ_{F_π} defined in Expression (4), and E_f is the subset of Ξ_{F_π} that maximizes the value of $\dim P_{E_f}$ if and only if there are no zero rows in $\mathbf{N}(\mathbf{D}_{E_f}^T)$.

Proof (sufficient part): the sufficiency of Theorem 3.1 has been proven by the following inverse negative proposition.

If there exist zero rows in $\mathbf{N}(\mathbf{D}_{E_f}^T)$, then E_f is not the element in Ξ_{F_π} that maximizes the value of $\dim P_{E_f}$.

Let $E_f = \{e_1, e_2, \dots, e_n\}$, $\mathbf{D}_E^T = [\mathbf{D}_{e_1}^T \ \mathbf{D}_{e_2}^T \ \dots \ \mathbf{D}_{e_n}^T]$; without loss of generality, we

can assume that the last m rows of $\mathbf{N}(\mathbf{D}_{E_f}^T)$ are zero rows.

$$\begin{aligned} \mathbf{N}(\mathbf{D}_{E_f}^T) &= [N_1 \quad N_2 \quad \dots \quad N_{n-r}] \\ &= \begin{bmatrix} N_{11} & N_{12} & \dots & N_{1(n-r)} \\ N_{21} & N_{22} & \dots & N_{2(n-r)} \\ \vdots & \vdots & \ddots & \vdots \\ N_{(n-m)1} & N_{(n-m)2} & \dots & N_{(n-m)(n-r)} \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \\ &= \begin{bmatrix} N_1^{\text{row}} \\ N_2^{\text{row}} \\ \vdots \\ N_{n-m}^{\text{row}} \\ N_{n-m+1}^{\text{row}} \\ \vdots \\ N_n^{\text{row}} \end{bmatrix}. \end{aligned}$$

According to $\mathbf{D}_{E_f}^T \mathbf{N}(\mathbf{D}_{E_f}^T) = \sum_{i=1}^n \mathbf{D}_{e_i}^T N_i^{\text{row}} = 0$, there is a one-to-one correspondence between N_i^{row} and $\mathbf{D}_{e_i}^T$; i.e., there is a one-to-one correspondence between the rows of $\mathbf{N}(\mathbf{D}_{E_f}^T)$ and the elements of E_f .

Let $E'_f = \{e_1, e_2, \dots, e_{n-m}\}$.

$$\begin{aligned} \mathbf{N}(\mathbf{D}_{E'_f}^T) &= [N_1 \quad N_2 \quad \dots \quad N_{n-r}] \\ &= \begin{bmatrix} N_{11} & N_{12} & \dots & N_{1(n-r)} \\ N_{21} & N_{22} & \dots & N_{2(n-r)} \\ \vdots & \vdots & \ddots & \vdots \\ N_{(n-m)1} & N_{(n-m)2} & \dots & N_{(n-m)(n-r)} \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \\ &= \begin{bmatrix} N_1^{\text{row}} \\ N_2^{\text{row}} \\ \vdots \\ N_{n-m}^{\text{row}} \\ N_{n-m+1}^{\text{row}} \\ \vdots \\ N_n^{\text{row}} \end{bmatrix} \end{aligned}$$

$[N_1 \quad N_2 \quad \dots \quad N_{n-r}]$ forms a basis for the null space of $\mathbf{D}_{E_f}^T$, implying that $[N'_1 \quad N'_2 \quad \dots \quad N'_{n-r}]$ forms a basis for the null space of $\mathbf{D}_{E'_f}^T$.

Next, we will prove that E'_f is an equality set, i.e.,

$$E'_f = \left\{ i \in \{1, \dots, q\} \mid \mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{y} = b_i, \forall (\mathbf{x}, \mathbf{y}) \in P_{E'_f} \right\}.$$

Since E_f is an equation set, we only need to prove that $\forall e_i \in E'_f, \exists (\mathbf{x}, \mathbf{y}) \in P_{E'_f}$, s.t. $\mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{y} \neq b_i$,

where

$$E'_f = \{e_{n-m+1}, e_{n-m+2}, \dots, e_n\}.$$

Since $\mathbf{D}_{E_f \setminus \{e_i\}}^T \mathbf{N}(\mathbf{D}_{E_f \setminus \{e_i\}}^T) = 0$, the null space of $\mathbf{D}_{E_f}^T$ and $\mathbf{D}_{E_f \setminus \{e_i\}}^T$ has the same number of bases, and thus

$$\text{rank } \mathbf{D}_{E_f}^T - \text{rank } \mathbf{D}_{E_f \setminus \{e_i\}}^T = 1. \tag{5}$$

Equation (5) implies that

$$\text{rank} [\mathbf{C}_{E_f} \mathbf{D}_{E_f}] - \text{rank} [\mathbf{C}_{E_f \setminus \{e_i\}} \mathbf{D}_{E_f \setminus \{e_i\}}] = 1. \tag{6}$$

Then, combining Proposition 2.5 with Expression (6), we get $\dim P_{E_f} = k + d - \text{rank} [\mathbf{C}_{E_f} \mathbf{D}_{E_f}] = \dim P_{E_f \setminus \{e_i\}} - 1$.

$E_{E_f \setminus \{e_i\}} \subset E_f$ implies that

$$P_{E_f \setminus \{e_i\}} \supseteq P_{E_f}. \tag{8}$$

Combining Expressions (7) and (8) gives us

$$P_{E_f \setminus \{e_i\}} \supset P_{E_f}.$$

Thus, $P_{E_f \setminus \{e_i\}} \setminus P_{E_f} \neq \emptyset$. Let $\mathbf{x} \in P_{E_f \setminus \{e_i\}} \setminus P_{E_f}$, $\mathbf{C}_i \mathbf{x} + \mathbf{D}_i \mathbf{y} \neq b_i$, and thus E'_f is an equality set. We can verify that

$$\mathbf{N}(\mathbf{D}_{E_f}^T)^T \mathbf{C}_{E_f} = \mathbf{N}(\mathbf{D}_{E'_f}^T)^T \mathbf{C}_{E'_f},$$

$$\mathbf{N}(\mathbf{D}_{E_f}^T)^T \mathbf{b}_{E_f} = \mathbf{N}(\mathbf{D}_{E'_f}^T)^T \mathbf{b}_{E'_f}.$$

From Proposition 2.4, $\text{aff } P_{E'_f} = \text{aff } P_{E_f}$, and thus $E'_f \in \Xi_{F_\pi}$. Therefore, $\dim P_{E'_f} \geq \dim P_{E_f \setminus \{e_i\}} > \dim P_{E_f}$, and the sufficient part is proven.

Necessary part: the necessary part is proven by contradiction.

Suppose that each row in $\mathbf{N}(\mathbf{D}_{E_f}^T)$ is not a zero row. If $\exists E_f^* \in \Xi_{F_\pi}$, then $\dim P_{E_f^*} > \dim P_{E_f}$. Let $\dim P_{E_f^*} - \dim P_{E_f} = n_c$, since the projections of E_f^* and E_f have the same affine hull. This can be obtained from Proposition 2.5:

$$\text{rank } \mathbf{D}_{E_f} - \text{rank } \mathbf{D}_{E_f^*} = n_c,$$

since $\pi_d(\text{aff } P_{E_f}) = \pi_d(\text{aff } P_{E_f^*})$. Proposition 2.4 implies that $\mathbf{N}(\mathbf{D}_{E_f}^T)$ and $\mathbf{N}(\mathbf{D}_{E_f^*}^T)$ have the same number of columns, i.e., $n^* - \text{rank } \mathbf{D}_{E_f^*} = n - \text{rank } \mathbf{D}_{E_f}$, $n - n^* = n_c$.

We can denote $\mathbf{N}(\mathbf{D}_{E_f^*}^T)$ as follows.

$$\begin{aligned} \mathbf{N}(\mathbf{D}_{E_f^*}^T) &= [N_1^* \quad N_2^* \quad \dots \quad N_{n-r}^*] \\ &= \begin{bmatrix} N_{11}^* & N_{12}^* & \dots & N_{1(n-r)}^* \\ N_{21}^* & N_{22}^* & \dots & N_{2(n-r)}^* \\ \vdots & \vdots & \ddots & \vdots \\ N_{(n^*)1} & N_{(n^*)2} & \dots & N_{(n^*)(n-r)} \end{bmatrix} \end{aligned}$$

Then, we add n_c zero elements to each column of $\mathbf{N}(\mathbf{D}_{E_f}^T)$ as follows.

$$\mathbf{N}(\mathbf{D}_{E_f}^T) = [N_1^* \quad N_2^* \quad \dots \quad N_{n-r}^*]$$

$$= \begin{bmatrix} N_{11}^* & N_{12}^* & \dots & N_{1(n-r)}^* \\ N_{21}^* & N_{22}^* & \dots & N_{2(n-r)}^* \\ \vdots & \vdots & \ddots & \vdots \\ N_{(n^*)1} & N_{(n^*)2} & \dots & N_{(n^*)(n-r)} \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix}$$

Since $N_1^*, N_2^*, \dots, N_{n-r}^*$ are linearly independent, $N_1^*, N_2^*, \dots, N_{n-r}^*$ are bases of $\mathbf{D}_{E_f}^T$. There are no zero rows in $\mathbf{N}(\mathbf{D}_{E_f}^T)$, and thus N_1, N_2, \dots, N_{n-r} cannot be expressed linearly by $N_1^*, N_2^*, \dots, N_{n-r}^*$, which is a contradiction. Therefore, the necessary part is proven.

Corollary 3.1 Let $E_f = \{e_1, e_2, \dots, e_n, \}$ $\in \Xi_{F_\pi}$. If there exist zero rows in $\mathbf{N}(\mathbf{D}_{E_f}^T)$, then E'_f , which consists of elements of E_f corresponding to non-zero rows in $\mathbf{N}(\mathbf{D}_{E_f}^T)$, is the element in Ξ_{F_π} that maximizes the value of $\dim P_{E_f}$.

Corollary 3.1 is a direct consequence of Theorem 3.1, and it implies that it is no longer necessary to discuss the uniqueness of the solutions in the SHOOT and ADJ oracles. However, it only takes an arbitrary solution as the unique solution and then gets the E_f that maximizes the value of $\dim P_{E_f}$ in Ξ_{F_π} .

3.3.1 Improved SHOOT oracle

Step 1 (selecting a direction γ and finding a point on the surface of P)

By randomly choosing $\gamma \in R^d$, we solve the following linear programming.

$$(r^*, y^*) = \arg \max_{(r, y) \in R \times R^k} \quad (9)$$

s.t. $\mathbf{C}\gamma r + \mathbf{D}y \leq \mathbf{b}$

Step 2 (whether a facet of $\pi_d(P)$ is found)

$$E_{f_0} = \{i | \mathbf{C}_i \gamma r + \mathbf{D}_i y = b_i\} \quad (10)$$

If $\text{Rank}(\mathbf{N}(\mathbf{D}_{E_{f_0}}^T) \mathbf{C}_{E_{f_0}}) = 1$, continue. Otherwise, go to Step 1.

Step 3 (ensuring the E_{f_0} is the element with the maximal $\dim P_{E_{f_0}}$ in Ξ_{F_π})

If there exist zero rows in $\mathbf{N}(\mathbf{D}_{E_{f_0}}^T)$, let $E_{f_0} \leftarrow E'_{f_0}$ as Corollary 3.1. Otherwise, continue.

Step 4 (computing the affine hull of the facet and normalizing)

$$[\mathbf{a}_f^T, b_f] = \mathbf{N}(\mathbf{D}_{E_{f_0}}^T)^T [\mathbf{C}_{E_{f_0}}, b_{E_{f_0}}]$$

$$[\mathbf{a}_f^T, b_f] = \frac{\text{sign} b_f}{\|\mathbf{a}_f\|_2} [\mathbf{a}_f^T, b_f]$$

3.3.2 Improved ADJ oracle

Step 1 (finding a point on the adjacent facet of $\pi_d(P)$ and its corresponding pre-image)

Solve the following linear programming.

$$(x^*, y^*) = \arg \max_{x, y} \mathbf{a}_r^T x$$

s.t. $\mathbf{C}_{E_r} x + \mathbf{D}_{E_r} y \leq b_{E_r}$,

$$\mathbf{a}_f^T x = b_f (1 - \varepsilon)$$

Step 2 (searching for the E_f of Ξ_{F_π} that maximizes the value of $\dim P_{E_f}$)

$$E_{\text{adj}} \leftarrow \{i \in E_r | \mathbf{C}_i x^* + \mathbf{D}_i y^* = b_i\}$$

If there exist zero rows in $\mathbf{N}(\mathbf{D}_{E_{\text{adj}}}^T)$, let $E_{\text{adj}} \leftarrow E'_{\text{adj}}$ as Corollary 3.1. Otherwise, continue.

Step 3 (computing the affine hull of the adjacent facet and normalizing)

$$[\mathbf{a}_{\text{adj}}^T, b_{\text{adj}}] = \mathbf{N}(\mathbf{D}_{E_{\text{adj}}}^T)^T [\mathbf{C}_{E_{\text{adj}}}, b_{E_{\text{adj}}}]$$

$$[\mathbf{a}_{\text{adj}}^T, b_{\text{adj}}] = \frac{\text{sign} b_{\text{adj}}}{\|\mathbf{a}_{\text{adj}}\|_2} [\mathbf{a}_{\text{adj}}^T, b_{\text{adj}}]$$

4 An improved strategy without searching for E_r

In this section, we first explain the obstacles to searching for E_r in the case of $\dim P_{E_f} > d - 1$, an improved strategy that does not require E_r is proposed, and then a new criterion is given to judge whether the two ridges are the same.

4.1 The barrier to searching for E_r

In the RDG oracle, if $\dim P_{E_f} > d - 1$, this requires a recursive call to the ESP algorithm that reduces the dimension onto which we are projecting at each step.

We note that \mathbf{a}_r and b_r can easily be obtained through this strategy, but it is often expensive to search for E_r . The only way to do this is to list all of the equality sets $E_c = \{1, 2, \dots, q\} \setminus E_f, \{E_p^i = E_f \cup E_i\}$, and $E_i \subset E_c$, and searching for the $\pi_d(E_p^i)$ is equal to $\{x | \tilde{\mathbf{a}}_f^T x = \tilde{b}_f\} \cap \pi_{d-1}(\tilde{P})$. The computational effort increases in an exponential manner in the number of dimensions.

4.2 New criteria without E_r

To improve the computational efficiency of the algorithm when the optimizer of LP is not unique, the search for E_r has been abandoned. However, in the original ESP algorithm, E_r is needed to judge the occurrence times of ridges, and hence we have proposed a new criterion to judge whether the two ridges in $\pi_d(P)$ are the same without E_r .

Lemma 4.1 Let P be a polytope. $F_1 = \{x | a_{f_1}^T x = b_{f_1}\} \cap \pi_d(P)$ and $F_2 = \{x | a_{f_2}^T x = b_{f_2}\} \cap \pi_d(P)$ are the facets of P , $R_1 = \{x | a_{r_1}^T x = b_{r_1}\} \cap F_1$ and $R_2 = \{x | a_{r_2}^T x = b_{r_2}\} \cap F_2$ are the ridges of P , and $R_1 \subset F_1$ and $R_2 \subset F_2$. Then, $F_1 \cap F_2 = R_1 = R_2$ if and only if

$$\text{rank} \begin{bmatrix} a_{f_1}^T \\ a_{f_2}^T \\ a_{r_1}^T \\ a_{r_2}^T \end{bmatrix} = \text{rank} \begin{bmatrix} a_{f_1}^T & b_{f_1} \\ a_{f_2}^T & b_{f_2} \\ a_{r_1}^T & b_{r_1} \\ a_{r_2}^T & b_{r_2} \end{bmatrix} = 2. \tag{11}$$

Proof Only the proof of sufficiency is given here, and the necessity is obvious. Since $a_{f_1} \perp a_{r_1}, a_{f_2} \perp a_{r_2}$,

$$\text{rank} \begin{bmatrix} a_{f_1}^T & b_{f_1} \\ a_{r_1}^T & b_{r_1} \end{bmatrix} = \text{rank} \begin{bmatrix} a_{f_2}^T & b_{f_2} \\ a_{r_2}^T & b_{r_2} \end{bmatrix} = 2. \tag{12}$$

Combining Expression (11) with Expression (12) gives us

$$\begin{aligned} & \{x | a_{r_1}^T x = b_{r_1}\} \cap \{x | a_{f_2}^T x = b_{f_2}\} \\ &= \{x | a_{r_2}^T x = b_{r_2}\} \cap \{x | a_{f_2}^T x = b_{f_2}\}. \end{aligned}$$

Note that

$$\begin{aligned} R_2 &= \{x | a_{r_2}^T x = b_{r_2}\} \cap \{x | a_{f_2}^T x = b_{f_2}\} \cap \pi_d(P), \\ R_1 &= \{x | a_{r_1}^T x = b_{r_1}\} \cap \{x | a_{f_1}^T x = b_{f_1}\} \cap \pi_d(P), \end{aligned}$$

and thus $R_1 = R_2$. Since the two facets of the polytope can only have one common ridge, $F_1 \cap F_2 = R_1 = R_2$.

4.3 Some other patches caused by the abandonment of searching for E_r

Since E_r is required in other steps in the original ESP algorithm, certain other patches need to be made.

In the ADJ oracle, LP (9) and Eq. (10) can be changed to LP (13) and Eq. (14), respectively.

$$\begin{aligned} (x^*, y^*) &= \arg \max_{x,y} a_r^T x \\ \text{s.t. } Cx + Dy &\leq b, \\ a_f^T x &= b_f (1 - \varepsilon) \end{aligned} \tag{13}$$

$$E_{\text{adj}} \leftarrow \{i \in \{1, 2, \dots, q\} | C_i x^* + D_i y^* = b_i\} \tag{14}$$

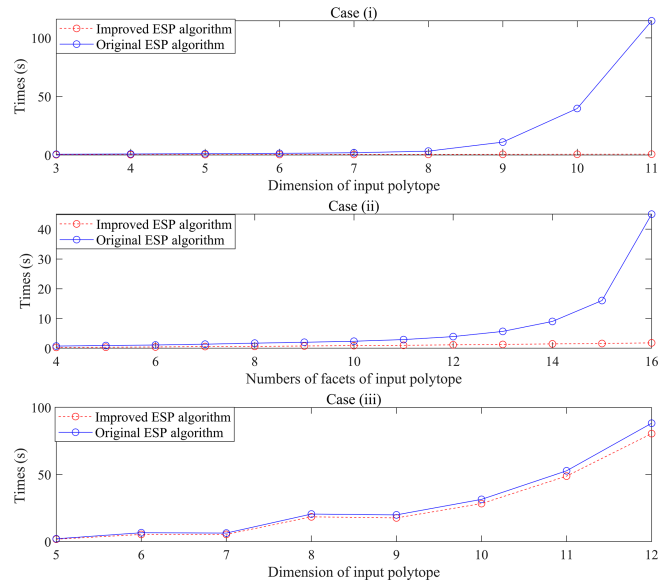


Figure 4. Comparative running times for typical cases.

We can verify that the modified algorithm still gets the correct result, though it also adds certain extra computation cost in the case of $\dim P_{E_f} = d - 1$. We note that $\dim(P_{E_f}) > d - 1$ if and only if the optimizer of LP is non-unique. A switching strategy can be adopted to take the original method until the optimizer of LP is non-unique.

5 Numerical simulations

In this section, the effectiveness of the proposed method is verified based on certain typical orthogonal projection cases and applications to N -step controllable set calculations in the control field.

5.1 Numerical simulations for the typical orthogonal projection cases

We select three cases for numerical simulation, and two of them are dual-degenerate, though one is not dual-degenerate.

- i. The projection of a three-dimensional n prism with $n+2$ half-spaces onto a two-dimensional plane
- ii. The projection of an n -dimensional cube with $2n$ half-spaces onto a two-dimensional plane
- iii. The projection of an n -dimensional cube onto a three-dimensional space

In cases (i) and (ii), the dual degeneracy occurs, whereas there is no dual degeneracy in case (iii). The simulation operation has been implemented in MATLAB 2019, and the running times of the original ESP and the improved ESP are shown in Fig. 4.

The running time of the original ESP algorithm increases exponentially with dimensions and the number of half-planes of the input polytopes under the dual-degeneracy cases, whereas the improved ESP algorithm significantly reduces the computational burden in the case of dual degeneracy. The running time is also optimized in the case of non-dual degeneracy, since it is no longer necessary to judge the uniqueness of linear programming solutions in the improved ESP algorithm.

5.2 Applications to N -step controllable set calculations

We apply the proposed method for the calculation of N -step controllable sets in the control field, and the problem setup is described as given below.

Consider the following linear discrete system.

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \underbrace{\begin{bmatrix} 1.1 & 2 \\ 0 & 0.95 \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ 0.0787 \end{bmatrix}}_{\mathbf{B}} \mathbf{u}$$

$\mathbf{x} = [x_1, x_2]^T$ and \mathbf{u} are the state and input vectors, respectively, subject to the constants as follows.

$$\underbrace{\begin{bmatrix} 0.125 & 0 \\ -0.125 & 0 \\ 0 & 0.125 \\ 0 & -0.125 \end{bmatrix}}_{\mathbf{A}_x} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \underbrace{\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\mathbf{A}_u} \mathbf{u} \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Then, we define the target set as follows.

$$\underbrace{\begin{bmatrix} 0.25 & 0 \\ -0.25 & 0 \\ 0 & 0.25 \\ 0 & -0.25 \end{bmatrix}}_{\mathbf{A}_f} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

The N -step controllable set $C(N)$ is defined as the collection of all initial states \mathbf{x}_0 that can evolve into the target set in N steps while satisfying the state and input constraints, and it can be transformed into the orthogonal projection problem as given below.

$$C(N) = \pi_d \left(P = \left\{ (\mathbf{x}, \mathbf{u}) \in R^d \times R^k \mid \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \leq \mathbf{b} \right\} \right),$$

where $d = 2, k = 1$,

$$\mathbf{D} = \begin{bmatrix} \mathbf{A}_u & 0 & \cdots & 0 \\ 0 & \mathbf{A}_u & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{A}_u \\ 0 & 0 & \cdots & 0 \\ \mathbf{A}_x \mathbf{B} & 0 & \cdots & 0 \\ \mathbf{A}_x \mathbf{A} \mathbf{B} & \mathbf{A}_x \mathbf{B} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_f \mathbf{A}^{N-1} \mathbf{B} & \mathbf{A}_f \mathbf{A}^{N-2} \mathbf{B} & \cdots & \mathbf{A}_f \mathbf{B} \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{A}_x \\ \mathbf{A}_x \mathbf{A} \\ \mathbf{A}_x \mathbf{A}^2 \\ \vdots \\ \mathbf{A}_f \mathbf{A}^N \end{bmatrix},$$

and

$$\mathbf{b} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}.$$

The N -step controllable set can be derived, as shown in Fig. 5, using either the improved ESP algorithm or the original ESP algorithm. When N is small, the region of the N -step controllable set changes significantly with the increasing N , though the trend of change gradually weakens and when $N \geq 12$. The N -step controllable set no longer changes with the increase in N .

Moreover, the running times of the original ESP and the improved ESP are shown in Fig. 6. When $N = 1, 2$, the two algorithms have similar computational efficiencies. When $N > 2$, the computational efficiency of the improved ESP method is higher than that of the original ESP method. This is due to the fact that, when $N = 1, 2$, dual degeneracy does not occur or occurs very rarely, while, when $N > 3$, dual degeneracy occurs frequently. Clearly, the improved ESP algorithm is better than the original ESP method in the application of N -step controllable set computation.

6 Conclusions

In this paper, the improved ESP algorithm becomes simpler, faster, and easier to implement in the case of dual degeneracy. This is achieved by the following methods. First, an alternative criterion has been proposed to directly obtain the

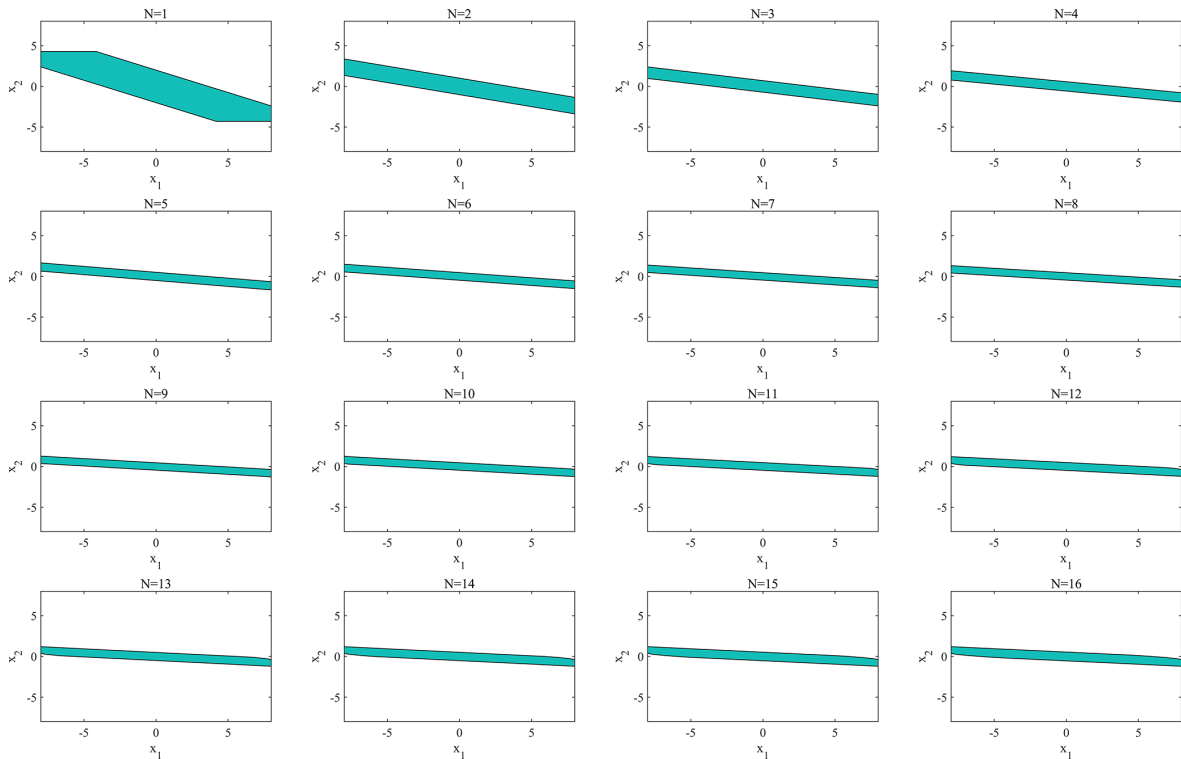


Figure 5. Results of the N -step controllable set.

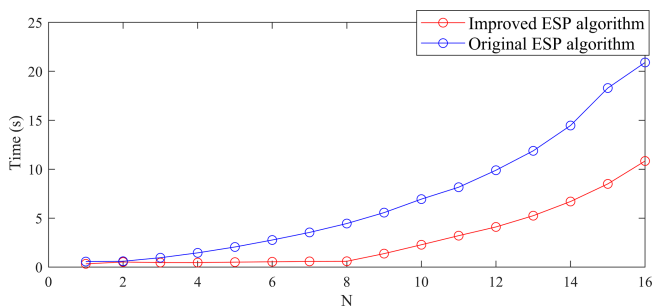


Figure 6. Comparative running times for the calculation of the N -step controllable set.

equality set that maximizes the value of $\dim P_{E_f}$. It avoids the uniqueness discussion of the optimizer in linear programming, which is not easy to implement. Further, it also reduces the operation times and simplifies the algorithm process. Second, an improved strategy has been presented to solve the difficulty in searching for the equality set E_r in the case of $\dim P_{E_f} > d - 1$, which significantly reduces the burden of computation. Finally, the effectiveness of the proposed improved ESP algorithm has been validated through numerical simulations.

Code availability. Software code in this study can be requested from the corresponding author.

Data availability. Research data in this study can be requested from the corresponding author.

Author contributions. BP and WX proposed improved strategies for the ESP algorithm. WX finished the simulation part. YL participated in the correction of the paper.

Competing interests. The contact author has declared that none of the authors has any competing interests.

Disclaimer. Publisher's note: Copernicus Publications remains neutral with regard to jurisdictional claims made in the text, published maps, institutional affiliations, or any other geographical representation in this paper. While Copernicus Publications makes every effort to include appropriate place names, the final responsibility lies with the authors.

Acknowledgements. The authors acknowledge the financial support of the National Natural Science Foundation of China. The authors would like to thank the editors and the anonymous reviewers

for their constructive comments and suggestions that have improved the quality of the paper.

Financial support. This research has been supported by the National Natural Science Foundation of China (grant no. 62103440).

Review statement. This paper was edited by Daniel Condurache and reviewed by two anonymous referees.

References

- Avis, D.: Computational experience with the reverse search vertex enumeration algorithm, *Optim. Method. Softw.*, 10, 107–124, <https://doi.org/10.1080/10556789808805706>, 1998.
- Balas, E.: Projection with a minimal system of inequalities, *Comput. Optim. Appl.*, 10, 189–193, <https://doi.org/10.1023/a:1018368920203>, 1998.
- Balas, E. and Oosten, M.: On the dimension of projected polyhedra, *Discrete Appl. Math.*, 87, 1–9, [https://doi.org/10.1016/s0166-218x\(98\)00096-1](https://doi.org/10.1016/s0166-218x(98)00096-1), 1998.
- Borrelli, F., Bemporad, A., and Morari, M.: *Predictive Control for Linear and Hybrid Systems*, Cambridge University Press, New York, <https://doi.org/10.1017/9781139061759>, 2017.
- Fukuda, K. and Prodon, A.: *Double description method revisited*, Springer, Berlin, https://doi.org/10.1007/3-540-61576-8_77, 1996.
- Jones, C. N.: *Polyhedral Tools for Control*, thesis, Department of Engineering University of Cambridge, Corpus ID: 117208061, 2005.
- Jones, C. N., Kerrigan, E. C., and Maciejowski, J. M.: Equality set projection: a new algorithm for the projection of polytopes in halfspace representation, Report CUED/F-INFENG/TR. 463, University of Cambridge, Corpus ID: 14790429, 2004.
- Jones, C. N., Kerrigan, E. C., and Maciejowski, J. M.: On polyhedral projection and parametric programming, *J. Optimiz. Theory App.*, 138, 207–220, <https://doi.org/10.1007/s10957-008-9384-4>, 2008.
- Keerthi, S. S. and Sridharan, K.: Solution of parametrized linear inequalities by Fourier elimination and its applications, *J. Optimiz. Theory App.*, 65, 161–169, <https://doi.org/10.1007/bf00941167>, 1990.
- Kwan, M., Sauermann, L., and Zhao, Y. F.: Extension complexity of low-dimensional polytopes, *T. Am. Math. Soc.*, 375, 4209–4250, <https://doi.org/10.1090/tran/8614>, 2022.
- Lee, Y. I. and Kouvaritakis, B.: Robust receding horizon predictive control for systems with uncertain dynamics and input saturation, *Automatica*, 36, 1497–1504, [https://doi.org/10.1016/s0005-1098\(00\)00064-9](https://doi.org/10.1016/s0005-1098(00)00064-9), 2000.
- Liu, H. Y., He, P. Q., Jiang, Y., Zhang, Q. L., and Xia, Q. Q.: A Simple Check Polytope Projection Penalized Algorithm for ADMM Decoding of LDPC Codes, *IEEE Access*, 11, 2524–2530, <https://doi.org/10.1109/access.2023.3234182>, 2023.
- Mangasarian, O. L.: Uniqueness of solution in linear programming, *Linear Algebra Appl.*, 25, 151–162, [https://doi.org/10.1016/0024-3795\(79\)90014-4](https://doi.org/10.1016/0024-3795(79)90014-4), 1979.
- Rakovic, S. V., Kerrigan, E. C., Mayne, D. Q., and Lygeros, J.: Reachability analysis of discrete-time systems with disturbances, *IEEE T. Automat. Contr.*, 51, 546–561, <https://doi.org/10.1109/tac.2006.872835>, 2006.
- Sadeghzadeh, A.: Gain-scheduled continuous-time control using polytope-bounded inexact scheduling parameters, *Int. J. Robust Nonlin.*, 28, 5557–5574, <https://doi.org/10.1002/rnc.4333>, 2018.
- Saraswat, V. and Hentenryck, V. P.: *Principles and Practice of Constraint Programming*, MIT Press, Boston, ISBN: 978-0-262-19361-0, 1995.
- Teissandier, D. and Delos, V.: Algorithm to calculate the Minkowski sums of 3-polytopes based on normal fans, *Comput. Aided Design*, 43, 1567–1576, <https://doi.org/10.1016/j.cad.2011.06.016>, 2011.
- Vincent, T. L. and Wu, Z. Y.: Estimating projections of the controllable set, *J. Guid. Control Dynam.*, 13, 572–575, <https://doi.org/10.2514/3.25372>, 1990.
- Webster, R. J.: *Convexity*, Oxford University Press, Oxford, <https://doi.org/10.1093/oso/9780198531470.001.0001>, 1994.
- Xia, Q. Q., Wang, X., Liu, H. J., and Zhang, Q. L.: A Hybrid Check Polytope Projection Algorithm for ADMM Decoding of LDPC Codes, *IEEE Commun. Lett.*, 25, 108–112, <https://doi.org/10.1109/lcomm.2020.3023600>, 2021.
- Xu, Y. and Guo, Q.: On Monotone Translation-Projection Covariant Minkowski Valuations, *Discrete Comput. Geom.*, 65, 713–729, <https://doi.org/10.1007/s00454-019-00069-y>, 2021.
- Ziegler, G. M.: *Lectures on Polytopes*, Springer, New York, <https://doi.org/10.1007/978-1-4613-8431-1>, 1995.