Combinatória Poliédrica
Projeções

Rafael C. S. Schouery
rafael@ic.unicamp.br

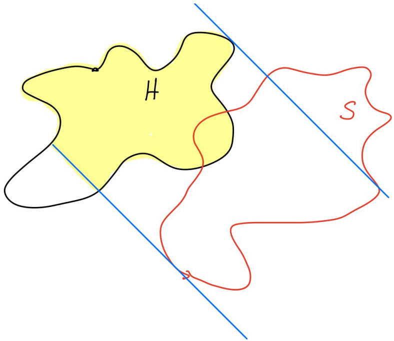
Universidade Estadual de Campinas

Atualizado em: 2023-09-21 21:13

Projeção

Projetando $S \subseteq \mathbb{R}^n$ sobre $H \subseteq \mathbb{R}^n$ na direção $\boldsymbol{c} \in \mathbb{R}^n$

$$\{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{x} \in H, \text{ existe } \lambda \in \mathbb{R} \text{ tal que } \boldsymbol{x} + \lambda \boldsymbol{c} \in S\}$$

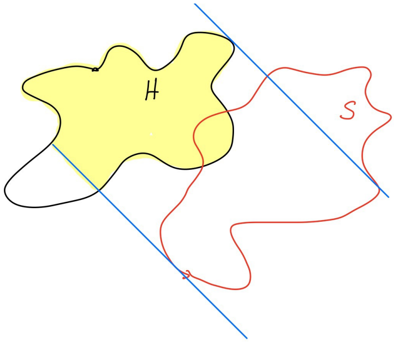


Projeção

Projetando $S \subseteq \mathbb{R}^n$ sobre $H \subseteq \mathbb{R}^n$ na direção $\boldsymbol{c} \in \mathbb{R}^n$

$$\{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{x} \in H, \text{ existe } \lambda \in \mathbb{R} \text{ tal que } \boldsymbol{x} + \lambda \boldsymbol{c} \in S\}$$

Isto é, $\boldsymbol{x} \in H$ está na projeção se é possível sair de $\boldsymbol{x}$ e encontrar um ponto de S ao andar na direção $\boldsymbol{c}$

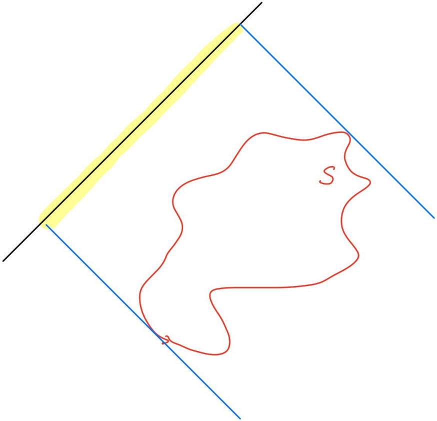


Projeção Ortogonal

Se H é o hiperplano $\{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{c}^T\boldsymbol{x} = \lambda\}$ então

$$\{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{x} \in H, \text{ existe } \lambda \in \mathbb{R} \text{ tal que } \boldsymbol{x} + \lambda\boldsymbol{c} \in S\}$$

é uma projeção ortogonal de S sobre H na direção $\boldsymbol{c}$



Exercício

Sejam $H \subseteq \mathbb{R}^n$ um subconjunto qualquer, $\boldsymbol{c} \in \mathbb{R}^n \setminus \{0\}$ e $H' = \{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{a}^T \boldsymbol{x} \leq \alpha\}$ um semi-espaço. Seja P_H a projeção de H' sobre H na direção $\boldsymbol{c}$. Prove que

(a) Se $\boldsymbol{a}$ é ortogonal a $\boldsymbol{c}$, então $P_H = H \cap H'$.

(b) Se $\boldsymbol{a}$ não é ortogonal a $\boldsymbol{c}$, então $P_H = H$.

Objetivo — Algoritmo de Projeção

Veremos um algoritmo para “calcular” a projeção de um poliedro $P(\boldsymbol{A}, \boldsymbol{b})$ sobre um conjunto (qualquer) H na direção $\boldsymbol{c}$.

Objetivo — Algoritmo de Projeção

Veremos um algoritmo para "calcular" a projeção de um poliedro $P(\boldsymbol{A}, \boldsymbol{b})$ sobre um conjunto (qualquer) H na direção $\boldsymbol{c}$.

A entrada do algoritmo é $\boldsymbol{A}$, $\boldsymbol{b}$ e $\boldsymbol{c}$

Objetivo — Algoritmo de Projeção

Veremos um algoritmo para "calcular" a projeção de um poliedro $P(\boldsymbol{A}, \boldsymbol{b})$ sobre um conjunto (qualquer) H na direção $\boldsymbol{c}$.

A entrada do algoritmo é $\boldsymbol{A}$, $\boldsymbol{b}$ e $\boldsymbol{c}$

A saída é uma matriz $\boldsymbol{D}$ e um vetor $\boldsymbol{d}$ tal que a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$ é exatemente $H \cup P(D, d)$

Objetivo — Algoritmo de Projeção

Veremos um algoritmo para "calcular" a projeção de um poliedro $P(\boldsymbol{A}, \boldsymbol{b})$ sobre um conjunto (qualquer) H na direção $\boldsymbol{c}$.

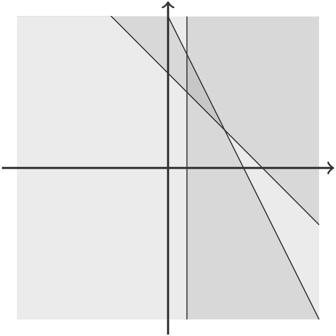
A entrada do algoritmo é $\boldsymbol{A}$, $\boldsymbol{b}$ e $\boldsymbol{c}$

A saída é uma matriz $\boldsymbol{D}$ e um vetor $\boldsymbol{d}$ tal que a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$ é exatemente $H \cup P(D, d)$

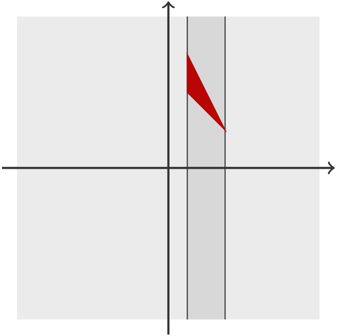
Isto é, $P(\boldsymbol{D}, \boldsymbol{d})$ "representa" qualquer projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ na direção $\boldsymbol{c}$, basta depois interceptar com H

Exemplo de $P(D, d)$

Projetando em $\boldsymbol{c} = (0, 1)$



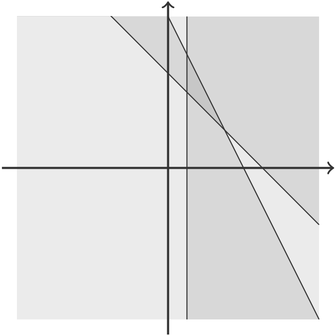
$$-1x_1 + 0x_2 \leq -1$$
$$-1x_1 - 1x_2 \leq -5$$
$$2x_1 + 1x_2 \leq 8$$

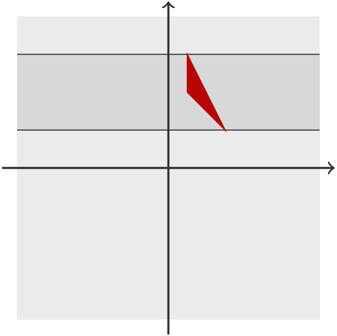


$$-1x_1 + 0x_2 \leq -1$$
$$1x_1 + 0x_2 \leq 3$$

Exemplo de $P(D, d)$

Projetando em $\boldsymbol{c} = (1, 0)$



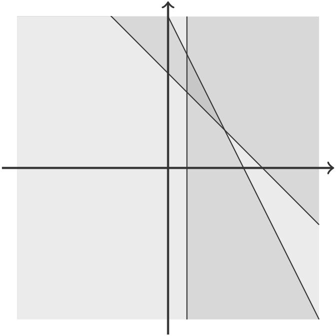


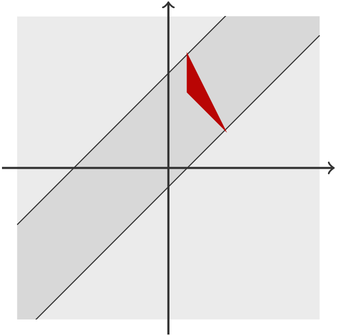
$$-1x_1 + 0x_2 \leq -1$$
$$-1x_1 - 1x_2 \leq -5$$
$$2x_1 + 1x_2 \leq 8$$

$$0x_1 + 1x_2 \leq 6$$
$$0x_1 - 1x_2 \leq -2$$

Exemplo de $P(D, d)$

Projetando em $\boldsymbol{c} = (1, 1)$





$$-1x_1 + 0x_2 \leq -1$$
$$-1x_1 - 1x_2 \leq -5$$
$$2x_1 + 1x_2 \leq 8$$

$$-1x_1 + 1x_2 \leq 5$$
$$1x_1 - 1x_2 \leq 1$$

Algoritmo de Projeção

Nosso objetivo é mostrar que o seguinte algoritmo está correto

procedure PROJEÇÃO($\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}$)
  Seja $M = \{1, \ldots, m\}$ o conjunto dos índices das linhas de $\boldsymbol{A}$
  Particione M em três partes $N = \{i \in M : \boldsymbol{A}_{i*}\boldsymbol{c} < 0\}$, $Z = \{i \in M : \boldsymbol{A}_{i*}\boldsymbol{c} = 0\}$, $P = \{i \in M : \boldsymbol{A}_{i*}\boldsymbol{c} > 0\}$
  Faça $r = |Z \cup (N \times P)|$
  Defina $R = 1, \ldots, r$
  Construa uma bijeção $p \colon R \to Z \cup (N \times P)$
  for $i = 1, \ldots, r$ **do**
    if $p(i) \in Z$ **then**
      $\boldsymbol{D}_{i*} = \boldsymbol{A}_{p(i)*}$
      $\boldsymbol{d}_i = \boldsymbol{b}_{p(i)}$
    else
      Seja $p(i) = (s, t) \in N \times P$
      $\boldsymbol{D}_{i*} = (\boldsymbol{A}_{t*}\boldsymbol{c})\boldsymbol{A}_{s*} - (\boldsymbol{A}_{s*}\boldsymbol{c})\boldsymbol{A}_{t*}$
      $\boldsymbol{d}_i = (\boldsymbol{A}_{t*}\boldsymbol{c})\boldsymbol{b}_s - (\boldsymbol{A}_{s*}\boldsymbol{c})\boldsymbol{b}_t$
  return $\boldsymbol{D}$ e $\boldsymbol{d}$

Olhando para dois semi-espaços

Seja $H_i = P(\boldsymbol{A}_{i*}, \boldsymbol{b}_i)$, sabemos que

$$P(\boldsymbol{A}, \boldsymbol{b}) = \bigcap_{i=1}^{m} H_i = \bigcap_{(i,j) \in M \times M} H_i \cap H_j$$

Olhando para dois semi-espaços

Seja $H_i = P(\boldsymbol{A}_{i*}, \boldsymbol{b}_i)$, sabemos que

$$P(\boldsymbol{A}, \boldsymbol{b}) = \bigcap_{i=1}^{m} H_i = \bigcap_{(i,j) \in M \times M} H_i \cap H_j$$

Primeiro, vamos mostrar que basta estudar como projetar a interseção de dois semiespaços numa direção $\boldsymbol{c}$ sobre um conjunto

Proposição

Proposição. Seja $P(\boldsymbol{A}, \boldsymbol{b})$ um poliedro em $\mathbb{R}^n$, $H \subseteq \mathbb{R}^n$, e $\boldsymbol{c} \in \mathbb{R}^n$. Então, a projeção P_H de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$ pode ser calculada da seguinte forma:

$$P_H = H \cap \bigcap_{(i,j) \in M \times M} H_{ij}$$

onde H_{ij} é a projeção de $H_i \cap H_j$ sobre $\mathbb{R}^n$ na direção $\boldsymbol{c}$.

Teorema — bem longo, então vamos por partes...

Teorema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a^T x \leq \alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{b}^T\boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

Teorema — bem longo, então vamos por partes...

Teorema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a^T x \leq \alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{b}^T\boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

(a) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ são ortogonais a $\boldsymbol{c}$, então $P_H = S_1 \cap S_2 \cap H$

Teorema — bem longo, então vamos por partes...

Teorema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a^T x \leq \alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{b}^T \boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

(a) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ são ortogonais a $\boldsymbol{c}$, então $P_H = S_1 \cap S_2 \cap H$

(b) Se um dos vetores $\boldsymbol{a}$, $\boldsymbol{b}$ é ortogonal a $\boldsymbol{c}$, digamos $\boldsymbol{a}^T \boldsymbol{c} = 0$ e $\boldsymbol{b}^T \boldsymbol{c} \neq 0$, então $P_H = S_1 \cap H$

Teorema — bem longo, então vamos por partes...

Teorema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a^T x \leq \alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{b}^T \boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

(a) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ são ortogonais a $\boldsymbol{c}$, então $P_H = S_1 \cap S_2 \cap H$

(b) Se um dos vetores $\boldsymbol{a}$, $\boldsymbol{b}$ é ortogonal a $\boldsymbol{c}$, digamos $\boldsymbol{a}^T \boldsymbol{c} = 0$ e $\boldsymbol{b}^T \boldsymbol{c} \neq 0$, então $P_H = S_1 \cap H$

(c) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ não são ortogonais a $\boldsymbol{c}$ e $\boldsymbol{a}^T \boldsymbol{c}$ e $\boldsymbol{b}^T \boldsymbol{c}$ tem o mesmo sinal, então $P_H = H$

Teorema — bem longo, então vamos por partes...

Teorema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^{\boldsymbol{T}}\boldsymbol{x} \leq \boldsymbol{\alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{b}^T\boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

(a) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ são ortogonais a $\boldsymbol{c}$, então $P_H = S_1 \cap S_2 \cap H$

(b) Se um dos vetores $\boldsymbol{a}$, $\boldsymbol{b}$ é ortogonal a $\boldsymbol{c}$, digamos $\boldsymbol{a}^T\boldsymbol{c} = 0$ e $\boldsymbol{b}^T\boldsymbol{c} \neq 0$, então $P_H = S_1 \cap H$

(c) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ não são ortogonais a $\boldsymbol{c}$ e $\boldsymbol{a}^T\boldsymbol{c}$ e $\boldsymbol{b}^T\boldsymbol{c}$ tem o mesmo sinal, então $P_H = H$

(d) Se $\boldsymbol{a}^T\boldsymbol{c} < 0$ e $\boldsymbol{b}^T\boldsymbol{c} > 0$, então tomando-se

$$\boldsymbol{d} = (\boldsymbol{b}^T\boldsymbol{c})\boldsymbol{a} - (\boldsymbol{a}^T\boldsymbol{c})\boldsymbol{b},$$

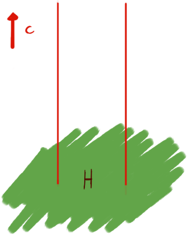
$$\delta = (\boldsymbol{b}^T\boldsymbol{c})\alpha - (\boldsymbol{a}^T\boldsymbol{c})\beta$$

temos que $P_H = H \cap \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{d}^T\boldsymbol{x} \leq \delta\}$. Adicionalmente, temos que a inequação $\boldsymbol{d}^T\boldsymbol{x} \leq \delta$ é uma combinação cônica das inequações $\boldsymbol{a}^T\boldsymbol{x} \leq \alpha$ e $\boldsymbol{b}^T\boldsymbol{x} \leq \beta$ e $\boldsymbol{d}$ é ortogonal a c.

Lema (a)

Lema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^T\boldsymbol{x} \leq \boldsymbol{\alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{b}^T\boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

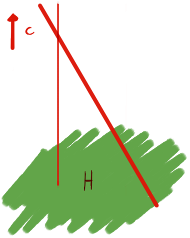
Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ são ortogonais a $\boldsymbol{c}$, então $P_H = S_1 \cap S_2 \cap H$



Lema (b)

Lema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{a}^T\boldsymbol{x} \leq \boldsymbol{\alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{b}^T\boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

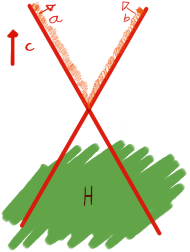
Se um dos vetores $\boldsymbol{a}$, $\boldsymbol{b}$ é ortogonal a $\boldsymbol{c}$, digamos $\boldsymbol{a}^T\boldsymbol{c} = 0$ e $\boldsymbol{b}^T\boldsymbol{c} \neq 0$, então $P_H = S_1 \cap H$



Lema (c)

Lema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{a^T x \leq \alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{b}^T \boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

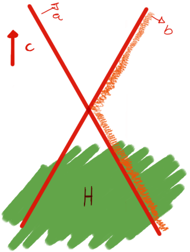
Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ não são ortogonais a $\boldsymbol{c}$ e $\boldsymbol{a}^T \boldsymbol{c}$ e $\boldsymbol{b}^T \boldsymbol{c}$ tem o mesmo sinal, então $P_H = H$



Lema (d)

Lema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n\colon \boldsymbol{a}^T\boldsymbol{x} \leq \boldsymbol{\alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n\colon \boldsymbol{b}^T\boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

Se $\boldsymbol{a}^T\boldsymbol{c} < 0$ e $\boldsymbol{b}^T\boldsymbol{c} > 0$, então tomando-se $\boldsymbol{d} = (\boldsymbol{b}^T\boldsymbol{c})\boldsymbol{a} - (\boldsymbol{a}^T\boldsymbol{c})\boldsymbol{b}$, $\delta = (\boldsymbol{b}^T\boldsymbol{c})\alpha - (\boldsymbol{a}^T\boldsymbol{c})\beta$ temos que $P_H = H \cap \{\boldsymbol{x} \in \mathbb{R}^n\colon \boldsymbol{d}^T\boldsymbol{x} \leq \delta\}$. Adicionalmente, temos que a inequação $\boldsymbol{d}^T\boldsymbol{x} \leq \delta$ é uma combinação cônica das inequações $\boldsymbol{a}^T\boldsymbol{x} \leq \alpha$ e $\boldsymbol{b}^T\boldsymbol{x} \leq \beta$ e $\boldsymbol{d}$ é ortogonal a c.



Teorema

Teorema. Sejam $\boldsymbol{c} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$, H um conjunto qualquer, $S_1 = \{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{a^T x} \leq \boldsymbol{\alpha}\}$ e $S_2 = \{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{b}^T\boldsymbol{x} \leq \beta\}$. Seja P_H a projeção de $S_1 \cap S_2$ sobre H na direção $\boldsymbol{c}$.

(a) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ são ortogonais a $\boldsymbol{c}$, então $P_H = S_1 \cap S_2 \cap H$

(b) Se um dos vetores $\boldsymbol{a}$, $\boldsymbol{b}$ é ortogonal a $\boldsymbol{c}$, digamos $\boldsymbol{a}^T\boldsymbol{c} = 0$ e $\boldsymbol{b}^T\boldsymbol{c} \neq 0$, então $P_H = S_1 \cap H$

(c) Se ambos os vetores $\boldsymbol{a}$, $\boldsymbol{b}$ não são ortogonais a $\boldsymbol{c}$ e $\boldsymbol{a}^T\boldsymbol{c}$ e $\boldsymbol{b}^T\boldsymbol{c}$ tem o mesmo sinal, então $P_H = H$

(d) Se $\boldsymbol{a}^T\boldsymbol{c} < 0$ e $\boldsymbol{b}^T\boldsymbol{c} > 0$, então tomando-se

$$\boldsymbol{d} = (\boldsymbol{b}^T\boldsymbol{c})\boldsymbol{a} - (\boldsymbol{a}^T\boldsymbol{c})\boldsymbol{b},$$
$$\delta = (\boldsymbol{b}^T\boldsymbol{c})\alpha - (\boldsymbol{a}^T\boldsymbol{c})\beta$$

temos que $P_H = H \cap \{\boldsymbol{x} \in \mathbb{R}^n \colon \boldsymbol{d}^T\boldsymbol{x} \leq \delta\}$. Adicionalmente, temos que a inequação $\boldsymbol{d}^T\boldsymbol{x} \leq \delta$ é uma combinação cônica das inequações $\boldsymbol{a}^T\boldsymbol{x} \leq \alpha$ e $\boldsymbol{b}^T\boldsymbol{x} \leq \beta$ e $\boldsymbol{d}$ é ortogonal a c.

Algoritmo de Projeção

procedure PROJEÇÃO($\boldsymbol{A}, \boldsymbol{b}, \boldsymbol{c}$)

  Seja $M = \{1, \ldots, m\}$ o conjunto dos índices das linhas de $\boldsymbol{A}$

  Particione M em três partes $N = \{i \in M : \boldsymbol{A}_{i*}\boldsymbol{c} < 0\}$, $Z = \{i \in M : \boldsymbol{A}_{i*}\boldsymbol{c} = 0\}$, $P = \{i \in M : \boldsymbol{A}_{i*}\boldsymbol{c} > 0\}$

  Faça $r = |Z \cup (N \times P)|$

  Defina $R = 1, \ldots, r$

  Construa uma bijeção $p \colon R \to Z \cup (N \times P)$

  for $i = 1, \ldots, r$ **do**

    if $p(i) \in Z$ **then**

      $\boldsymbol{D}_{i*} = \boldsymbol{A}_{p(i)*}$

      $\boldsymbol{d}_i = \boldsymbol{b}_{p(i)}$

    else

      Seja $p(i) = (s, t) \in N \times P$

      $\boldsymbol{D}_{i*} = (\boldsymbol{A}_{t*}\boldsymbol{c})\boldsymbol{A}_{s*} - (\boldsymbol{A}_{s*}\boldsymbol{c})\boldsymbol{A}_{t*}$

      $\boldsymbol{d}_i = (\boldsymbol{A}_{t*}\boldsymbol{c})\boldsymbol{b}_s - (\boldsymbol{A}_{s*}\boldsymbol{c})\boldsymbol{b}_t$

  return $\boldsymbol{D}$ e $\boldsymbol{d}$

Teorema (mais longo ainda)

Teorema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{Dx} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

Teorema (mais longo ainda)

Teorema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{Dx} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(a) Cada inequação $\boldsymbol{D}_{i*}\boldsymbol{x} \leq \boldsymbol{d}_i$ é combinação linear não-negativa do sistema $\boldsymbol{Ax} \leq \boldsymbol{b}$, i.e., para cada $i \in \{1, \ldots, r\}$, existe um vetor $\boldsymbol{u} \geq 0$ com $\boldsymbol{D}_{i*} = \boldsymbol{u}^T\boldsymbol{A}$ e $\boldsymbol{d}_i = \boldsymbol{u}^T\boldsymbol{b}$.

Teorema (mais longo ainda)

Teorema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{Dx} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(a) Cada inequação $\boldsymbol{D}_{i*}\boldsymbol{x} \leq \boldsymbol{d}_i$ é combinação linear não-negativa do sistema $\boldsymbol{Ax} \leq \boldsymbol{b}$, i.e., para cada $i \in \{1, \ldots, r\}$, existe um vetor $\boldsymbol{u} \geq 0$ com $\boldsymbol{D}_{i*} = \boldsymbol{u}^T\boldsymbol{A}$ e $\boldsymbol{d}_i = \boldsymbol{u}^T\boldsymbol{b}$.

(b) $\boldsymbol{D}_{i*}\boldsymbol{c} = 0$ para todo $i = 1, \ldots, r$.

Teorema (mais longo ainda)

Teorema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{D}\boldsymbol{x} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(a) Cada inequação $\boldsymbol{D}_{i*}\boldsymbol{x} \leq \boldsymbol{d}_i$ é combinação linear não-negativa do sistema $\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$, i.e., para cada $i \in \{1, \ldots, r\}$, existe um vetor $\boldsymbol{u} \geq 0$ com $\boldsymbol{D}_{i*} = \boldsymbol{u}^T\boldsymbol{A}$ e $\boldsymbol{d}_i = \boldsymbol{u}^T\boldsymbol{b}$.

(b) $\boldsymbol{D}_{i*}\boldsymbol{c} = 0$ para todo $i = 1, \ldots, r$.

(c) $P_H = H \cap P(\boldsymbol{D}, \boldsymbol{d})$

Teorema (mais longo ainda)

Teorema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{Dx} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(a) Cada inequação $\boldsymbol{D}_{i*}\boldsymbol{x} \leq \boldsymbol{d}_i$ é combinação linear não-negativa do sistema $\boldsymbol{Ax} \leq \boldsymbol{b}$, i.e., para cada $i \in \{1, \ldots, r\}$, existe um vetor $\boldsymbol{u} \geq 0$ com $\boldsymbol{D}_{i*} = \boldsymbol{u}^T\boldsymbol{A}$ e $\boldsymbol{d}_i = \boldsymbol{u}^T\boldsymbol{b}$.

(b) $\boldsymbol{D}_{i*}\boldsymbol{c} = 0$ para todo $i = 1, \ldots, r$.

(c) $P_H = H \cap P(\boldsymbol{D}, \boldsymbol{d})$

(d) Tome $\boldsymbol{x}' \in H$ e N, Z e P definidos como no algoritmo. Defina

$$\lambda_i = \frac{1}{\boldsymbol{A}_{i*}\boldsymbol{c}}(\boldsymbol{b}_i - \boldsymbol{A}_{i*}\boldsymbol{x}') \quad \text{para todo } i \in P \cup N$$

$$L = \begin{cases} -\infty & \text{se } N = \emptyset \\ \max\{\lambda_i | i \in N\} & \text{caso contrário} \end{cases}$$

$$U = \begin{cases} +\infty & \text{se } P = \emptyset \\ \min\{\lambda_i | i \in P\} & \text{caso contrário} \end{cases}$$

Então,

Teorema (mais longo ainda)

Teorema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{D}\boldsymbol{x} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A},\boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(a) Cada inequação $\boldsymbol{D}_{i*}\boldsymbol{x} \leq \boldsymbol{d}_i$ é combinação linear não-negativa do sistema $\boldsymbol{A}\boldsymbol{x} \leq \boldsymbol{b}$, i.e., para cada $i \in \{1,\ldots,r\}$, existe um vetor $\boldsymbol{u} \geq 0$ com $\boldsymbol{D}_{i*} = \boldsymbol{u}^T\boldsymbol{A}$ e $\boldsymbol{d}_i = \boldsymbol{u}^T\boldsymbol{b}$.

(b) $\boldsymbol{D}_{i*}\boldsymbol{c} = 0$ para todo $i = 1,\ldots,r$.

(c) $P_H = H \cap P(\boldsymbol{D},\boldsymbol{d})$

(d) Tome $\boldsymbol{x}' \in H$ e N, Z e P definidos como no algoritmo. Defina

$$\lambda_i = \frac{1}{\boldsymbol{A}_{i*}\boldsymbol{c}}(\boldsymbol{b}_i - \boldsymbol{A}_{i*}\boldsymbol{x}') \quad \text{para todo } i \in P \cup N$$

$$L = \begin{cases} -\infty & \text{se } N = \emptyset \\ \max\{\lambda_i | i \in N\} & \text{caso contrário} \end{cases}$$

$$U = \begin{cases} +\infty & \text{se } P = \emptyset \\ \min\{\lambda_i | i \in P\} & \text{caso contrário} \end{cases}$$

Então,

(d1) $\boldsymbol{x}' \in P(\boldsymbol{D},\boldsymbol{d})$ implica que $L \leq U$ e $\boldsymbol{x}' + \lambda\boldsymbol{c} \in P(\boldsymbol{A},\boldsymbol{b})$ para todo λ tal que $L \leq \lambda \leq U$.

Teorema (mais longo ainda)

Teorema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{Dx} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(a) Cada inequação $\boldsymbol{D}_{i*}\boldsymbol{x} \leq \boldsymbol{d}_i$ é combinação linear não-negativa do sistema $\boldsymbol{Ax} \leq \boldsymbol{b}$, i.e., para cada $i \in \{1, \ldots, r\}$, existe um vetor $\boldsymbol{u} \geq 0$ com $\boldsymbol{D}_{i*} = \boldsymbol{u}^T\boldsymbol{A}$ e $\boldsymbol{d}_i = \boldsymbol{u}^T\boldsymbol{b}$.

(b) $\boldsymbol{D}_{i*}\boldsymbol{c} = 0$ para todo $i = 1, \ldots, r$.

(c) $P_H = H \cap P(\boldsymbol{D}, \boldsymbol{d})$

(d) Tome $\boldsymbol{x}' \in H$ e N, Z e P definidos como no algoritmo. Defina

$$\lambda_i = \frac{1}{\boldsymbol{A}_{i*}\boldsymbol{c}}(\boldsymbol{b}_i - \boldsymbol{A}_{i*}\boldsymbol{x}') \quad \text{para todo } i \in P \cup N$$

$$L = \begin{cases} -\infty & \text{se } N = \emptyset \\ \max\{\lambda_i | i \in N\} & \text{caso contrário} \end{cases}$$

$$U = \begin{cases} +\infty & \text{se } P = \emptyset \\ \min\{\lambda_i | i \in P\} & \text{caso contrário} \end{cases}$$

Então,

(d1) $\boldsymbol{x}' \in P(\boldsymbol{D}, \boldsymbol{d})$ implica que $L \leq U$ e $\boldsymbol{x}' + \lambda\boldsymbol{c} \in P(\boldsymbol{A}, \boldsymbol{b})$ para todo λ tal que $L \leq \lambda \leq U$.
(d2) $\boldsymbol{x}' + \lambda\boldsymbol{c} \in P(\boldsymbol{A}, \boldsymbol{b})$ implica que $L \leq \lambda \leq U$ e $\boldsymbol{x}' \in P(\boldsymbol{D}, \boldsymbol{d})$.

Lema (a)

Lema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{Dx} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(a) Cada inequação $\boldsymbol{D}_{i*}\boldsymbol{x} \leq \boldsymbol{d}_i$ é combinação linear não-negativa do sistema $\boldsymbol{Ax} \leq \boldsymbol{b}$, i.e., para cada $i \in \{1, \ldots, r\}$, existe um vetor $\boldsymbol{u} \geq 0$ com $\boldsymbol{D}_{i*} = \boldsymbol{u}^T\boldsymbol{A}$ e $\boldsymbol{d}_i = \boldsymbol{u}^T\boldsymbol{b}$.

Lema (b)

Lema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{D}\boldsymbol{x} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(b) $\boldsymbol{D}_{i*}\boldsymbol{c} = 0$ para todo $i = 1, \ldots, r$.

Lema (d)

Lema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{D}\boldsymbol{x} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(d) Tome $\boldsymbol{x}' \in H$ e N, Z e P definidos como no algoritmo. Defina

$$\lambda_i = \frac{1}{\boldsymbol{A}_{i*}\boldsymbol{c}}(\boldsymbol{b}_i - \boldsymbol{A}_{i*}\boldsymbol{x}') \quad \text{para todo } i \in P \cup N$$

$$L = \begin{cases} -\infty & \text{se } N = \emptyset \\ \max\{\lambda_i | i \in N\} & \text{caso contrário} \end{cases}$$

$$U = \begin{cases} +\infty & \text{se } P = \emptyset \\ \min\{\lambda_i | i \in P\} & \text{caso contrário} \end{cases}$$

Então,

(d1) $\boldsymbol{x}' \in P(\boldsymbol{D}, \boldsymbol{d})$ implica que $L \leq U$ e $\boldsymbol{x}' + \lambda\boldsymbol{c} \in P(\boldsymbol{A}, \boldsymbol{b})$ para todo λ tal que $L \leq \lambda \leq U$.

(d2) $\boldsymbol{x}' + \lambda\boldsymbol{c} \in P(\boldsymbol{A}, \boldsymbol{b})$ implica que $L \leq \lambda \leq U$ e $\boldsymbol{x}' \in P(\boldsymbol{D}, \boldsymbol{d})$.

Lema (c)

Lema. Sejam $\boldsymbol{A} \in \mathbb{R}^{m \times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, $\boldsymbol{c} \in \mathbb{R}^n$ e $H \subseteq \mathbb{R}^n$ um conjunto qualquer. Considere o sistema $\boldsymbol{D}\boldsymbol{x} \leq \boldsymbol{d}$ construído aplicando-se o algoritmo de projeção e seja P_H a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$. Então,

(c) $P_H = H \cap P(\boldsymbol{D}, \boldsymbol{d})$

Projeção de Poliedro sobre Poliedro

Proposição. Se $H = P(\boldsymbol{A}', \boldsymbol{b}')$ é um poliedro, então a projeção P_H de $P(\boldsymbol{A}, \boldsymbol{b})$ sobre H na direção $\boldsymbol{c}$ é um poliedro.

Eliminação de Fourier-Motzkin

Podemos "eliminar" a variável $\boldsymbol{x}_j$ projetando sobre o hiperplano $\{x \in \mathbb{R}^n \colon x_j = 0\}$. A projeção é $\{x \in P(D,d) \colon x_j = 0\}$.

procedure $\textsc{FourierMotzkin}(\boldsymbol{A}, \boldsymbol{b}, j)$
    Seja $M = \{1, \ldots, m\}$ o conjunto dos índices das linhas de $\boldsymbol{A}$
    Particione M em três partes $N = \{i \in M \colon \boldsymbol{A}_{ij} < 0\}$, $Z = \{i \in M \colon \boldsymbol{A}_{ij} = 0\}$, $P = \{i \in M \colon \boldsymbol{A}_{ij} > 0\}$
    Faça $r = |Z \cup (N \times P)|$
    Defina $R = 1, \ldots, r$
    Construa uma bijeção $p \colon R \to Z \cup (N \times P)$
    for $i = 1, \ldots, r$ **do**
        if $p(i) \in Z$ **then**
            $\boldsymbol{D}_{i*} = \boldsymbol{A}_{p(i)*}$
            $\boldsymbol{d}_i = \boldsymbol{b}_{p(i)}$
        else
            seja $p(i) = (s,t) \in N \times P$
            $\boldsymbol{D}_{i*} = \boldsymbol{A}_{tj}\boldsymbol{A}_{s*} - \boldsymbol{A}_{sj}\boldsymbol{A}_{t*}$
            $\boldsymbol{d}_i = \boldsymbol{A}_{tj}\boldsymbol{b}_s - \boldsymbol{A}_{sj}\boldsymbol{b}_t$
    return $\boldsymbol{D}$ e $\boldsymbol{d}$

Proposição

Proposição. Seja $P(\boldsymbol{A}, \boldsymbol{b})$ um poliedro e $P(\boldsymbol{D}, \boldsymbol{d})$ o poliedro obtido aplicando o algoritmo de projeção descrito acima. Então,

$$P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset \text{ se e somente se } P(\boldsymbol{D}, \boldsymbol{d}) \neq \emptyset.$$

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

(i) $\boldsymbol{D}_1\boldsymbol{e}_1 = \boldsymbol{0}$ (pelo teorema)

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

(i) $\boldsymbol{D}_1\boldsymbol{e}_1 = \boldsymbol{0}$ (pelo teorema)

(ii) existe $\boldsymbol{U}_1 \geq \boldsymbol{0}$ tal que $\boldsymbol{U}_1\boldsymbol{A} = \boldsymbol{D}_1$ e $\boldsymbol{U}_1\boldsymbol{b} = \boldsymbol{d}_1$ (pelo teorema)

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

(i) $\boldsymbol{D}_1 \boldsymbol{e}_1 = \mathbf{0}$ (pelo teorema)

(ii) existe $\boldsymbol{U}_1 \geq \mathbf{0}$ tal que $\boldsymbol{U}_1 \boldsymbol{A} = \boldsymbol{D}_1$ e $\boldsymbol{U}_1 \boldsymbol{b} = \boldsymbol{d}_1$ (pelo teorema)

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

(i) $\boldsymbol{D}_1\boldsymbol{e}_1 = \mathbf{0}$ (pelo teorema)

(ii) existe $\boldsymbol{U}_1 \geq \mathbf{0}$ tal que $\boldsymbol{U}_1\boldsymbol{A} = \boldsymbol{D}_1$ e $\boldsymbol{U}_1\boldsymbol{b} = \boldsymbol{d}_1$ (pelo teorema)

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

(i) $\boldsymbol{D}_1\boldsymbol{e}_1 = \mathbf{0}$ (pelo teorema)

(ii) existe $\boldsymbol{U}_1 \geq \mathbf{0}$ tal que $\boldsymbol{U}_1\boldsymbol{A} = \boldsymbol{D}_1$ e $\boldsymbol{U}_1\boldsymbol{b} = \boldsymbol{d}_1$ (pelo teorema)

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

(i) $\boldsymbol{D}_1\boldsymbol{e}_1 = \mathbf{0}$ (pelo teorema)

(ii) existe $\boldsymbol{U}_1 \geq \mathbf{0}$ tal que $\boldsymbol{U}_1\boldsymbol{A} = \boldsymbol{D}_1$ e $\boldsymbol{U}_1\boldsymbol{b} = \boldsymbol{d}_1$ (pelo teorema)

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2 \geq \mathbf{0}$ tal que $\boldsymbol{U}_2\boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2\boldsymbol{d}_1 = \boldsymbol{d}_2$

Algoritmo para ver se o poliedro é vazio

Se eu aplicar a eliminação de Fourier-Motzkin em $\boldsymbol{x}_1$, eu obtenho $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$ tal que

(i) $\boldsymbol{D}_1 \boldsymbol{e}_1 = \mathbf{0}$ (pelo teorema)

(ii) existe $\boldsymbol{U}_1 \geq \mathbf{0}$ tal que $\boldsymbol{U}_1 \boldsymbol{A} = \boldsymbol{D}_1$ e $\boldsymbol{U}_1 \boldsymbol{b} = \boldsymbol{d}_1$ (pelo teorema)

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2 \boldsymbol{e}_2 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2 \geq \mathbf{0}$ tal que $\boldsymbol{U}_2 \boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2 \boldsymbol{d}_1 = \boldsymbol{d}_2$

(iii) $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_2, \boldsymbol{d}_2) \neq \emptyset$

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2 \geq \mathbf{0}$ tal que $\boldsymbol{U}_2\boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2\boldsymbol{d}_1 = \boldsymbol{d}_2$

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \boldsymbol{0}$

(ii) existe $\boldsymbol{U}_2 \geq \boldsymbol{0}$ tal que $\boldsymbol{U}_2\boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2\boldsymbol{d}_1 = \boldsymbol{d}_2$

(iii) $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_2, \boldsymbol{d}_2) \neq \emptyset$

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \boldsymbol{0}$

(ii) existe $\boldsymbol{U}_2 \geq \boldsymbol{0}$ tal que $\boldsymbol{U}_2\boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2\boldsymbol{d}_1 = \boldsymbol{d}_2$

(iii) $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_2, \boldsymbol{d}_2) \neq \emptyset$

Reescrevendo considerando $\boldsymbol{U}'_2 = \boldsymbol{U}_2\boldsymbol{U}_1$, temos

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2 \geq \mathbf{0}$ tal que $\boldsymbol{U}_2\boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2\boldsymbol{d}_1 = \boldsymbol{d}_2$

(iii) $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_2, \boldsymbol{d}_2) \neq \emptyset$

Reescrevendo considerando $\boldsymbol{U}_2' = \boldsymbol{U}_2\boldsymbol{U}_1$, temos

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$ e $\boldsymbol{D}_2\boldsymbol{e}_1 = \boldsymbol{U}_2\boldsymbol{D}_1\boldsymbol{e}_1 = \mathbf{0}$

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2 \geq \mathbf{0}$ tal que $\boldsymbol{U}_2\boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2\boldsymbol{d}_1 = \boldsymbol{d}_2$

(iii) $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_2, \boldsymbol{d}_2) \neq \emptyset$

Reescrevendo considerando $\boldsymbol{U}_2' = \boldsymbol{U}_2\boldsymbol{U}_1$, temos

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$ e $\boldsymbol{D}_2\boldsymbol{e}_1 = \boldsymbol{U}_2\boldsymbol{D}_1\boldsymbol{e}_1 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2' \geq \mathbf{0}$ tal que $\boldsymbol{U}_2'\boldsymbol{A} = \boldsymbol{D}_2$ e $\boldsymbol{U}_2'\boldsymbol{b} = \boldsymbol{d}_2$

Algoritmo para ver se o poliedro é vazio

Aplicando em $\boldsymbol{x}_2$ para $\boldsymbol{D}_1$ e $\boldsymbol{d}_1$, obtemos $\boldsymbol{D}_2$ e $\boldsymbol{d}_2$ tal que

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2 \geq \mathbf{0}$ tal que $\boldsymbol{U}_2\boldsymbol{D}_1 = \boldsymbol{D}_2$ e $\boldsymbol{U}_2\boldsymbol{d}_1 = \boldsymbol{d}_2$

(iii) $P(\boldsymbol{D}_1, \boldsymbol{d}_1) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_2, \boldsymbol{d}_2) \neq \emptyset$

Reescrevendo considerando $\boldsymbol{U}_2' = \boldsymbol{U}_2\boldsymbol{U}_1$, temos

(i) $\boldsymbol{D}_2\boldsymbol{e}_2 = \mathbf{0}$ e $\boldsymbol{D}_2\boldsymbol{e}_1 = \boldsymbol{U}_2\boldsymbol{D}_1\boldsymbol{e}_1 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_2' \geq \mathbf{0}$ tal que $\boldsymbol{U}_2'\boldsymbol{A} = \boldsymbol{D}_2$ e $\boldsymbol{U}_2'\boldsymbol{b} = \boldsymbol{d}_2$

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_2, \boldsymbol{d}_2) \neq \emptyset$

Algoritmo para ver se o poliedro é vazio

Após n passos:

Algoritmo para ver se o poliedro é vazio

Após n passos:

(i) $\boldsymbol{D}_n\boldsymbol{e}_n = \boldsymbol{D}_n\boldsymbol{e}_{n-1} = \cdots = \boldsymbol{D}_n\boldsymbol{e}_1 = \boldsymbol{0}$

Algoritmo para ver se o poliedro é vazio

Após n passos:

(i) $\boldsymbol{D}_n\boldsymbol{e}_n = \boldsymbol{D}_n\boldsymbol{e}_{n-1} = \cdots = \boldsymbol{D}_n\boldsymbol{e}_1 = \boldsymbol{0}$

(ii) existe $\boldsymbol{U}_n' \geq \boldsymbol{0}$ tal que $\boldsymbol{U}_n'\boldsymbol{A} = \boldsymbol{D}_n$ e $\boldsymbol{U}_n'\boldsymbol{b} = \boldsymbol{d}_n$

Algoritmo para ver se o poliedro é vazio

Após n passos:

(i) $\boldsymbol{D}_n\boldsymbol{e}_n = \boldsymbol{D}_n\boldsymbol{e}_{n-1} = \cdots = \boldsymbol{D}_n\boldsymbol{e}_1 = \boldsymbol{0}$

(ii) existe $\boldsymbol{U}_n' \geq \boldsymbol{0}$ tal que $\boldsymbol{U}_n'\boldsymbol{A} = \boldsymbol{D}_n$ e $\boldsymbol{U}_n'\boldsymbol{b} = \boldsymbol{d}_n$

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_n, \boldsymbol{d}_n) \neq \emptyset$

Algoritmo para ver se o poliedro é vazio

Após n passos:

(i) $\boldsymbol{D}_n \boldsymbol{e}_n = \boldsymbol{D}_n \boldsymbol{e}_{n-1} = \cdots = \boldsymbol{D}_n \boldsymbol{e}_1 = \mathbf{0}$

(ii) existe $\boldsymbol{U}_n' \geq \mathbf{0}$ tal que $\boldsymbol{U}_n' \boldsymbol{A} = \boldsymbol{D}_n$ e $\boldsymbol{U}_n' \boldsymbol{b} = \boldsymbol{d}_n$

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_n, \boldsymbol{d}_n) \neq \emptyset$

Por (i), $\boldsymbol{D}_n = \mathbf{0}$ pois $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\}$ é uma base do $\mathbb{R}^n$.

Algoritmo para ver se o poliedro é vazio

Após n passos:

(i) $\boldsymbol{D}_n\boldsymbol{e}_n = \boldsymbol{D}_n\boldsymbol{e}_{n-1} = \cdots = \boldsymbol{D}_n\boldsymbol{e}_1 = \boldsymbol{0}$

(ii) existe $\boldsymbol{U}_n' \geq \boldsymbol{0}$ tal que $\boldsymbol{U}_n'\boldsymbol{A} = \boldsymbol{D}_n$ e $\boldsymbol{U}_n'\boldsymbol{b} = \boldsymbol{d}_n$

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_n, \boldsymbol{d}_n) \neq \emptyset$

Por (i), $\boldsymbol{D}_n = \boldsymbol{0}$ pois $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\}$ é uma base do $\mathbb{R}^n$.

Logo, $P(\boldsymbol{D}_n, \boldsymbol{d}_n)$ é vazio se e somente se alguma coordenada de $\boldsymbol{d}_n$ é negativa

Algoritmo para ver se o poliedro é vazio

Após n passos:

(i) $\boldsymbol{D}_n\boldsymbol{e}_n = \boldsymbol{D}_n\boldsymbol{e}_{n-1} = \cdots = \boldsymbol{D}_n\boldsymbol{e}_1 = \mathbf{0}$

(ii) existe $\boldsymbol{U}'_n \geq \mathbf{0}$ tal que $\boldsymbol{U}'_n\boldsymbol{A} = \boldsymbol{D}_n$ e $\boldsymbol{U}'_n\boldsymbol{b} = \boldsymbol{d}_n$

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_n, \boldsymbol{d}_n) \neq \emptyset$

Por (i), $\boldsymbol{D}_n = \mathbf{0}$ pois $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\}$ é uma base do $\mathbb{R}^n$.

Logo, $P(\boldsymbol{D}_n, \boldsymbol{d}_n)$ é vazio se e somente se alguma coordenada de $\boldsymbol{d}_n$ é negativa

Idem para $P(\boldsymbol{A}, \boldsymbol{b})$

Algoritmo para ver se o poliedro é vazio

Após n passos:

(i) $\boldsymbol{D}_n \boldsymbol{e}_n = \boldsymbol{D}_n \boldsymbol{e}_{n-1} = \cdots = \boldsymbol{D}_n \boldsymbol{e}_1 = \boldsymbol{0}$

(ii) existe $\boldsymbol{U}'_n \geq \boldsymbol{0}$ tal que $\boldsymbol{U}'_n \boldsymbol{A} = \boldsymbol{D}_n$ e $\boldsymbol{U}'_n \boldsymbol{b} = \boldsymbol{d}_n$

(iii) $P(\boldsymbol{A}, \boldsymbol{b}) \neq \emptyset$ se e somente se $P(\boldsymbol{D}_n, \boldsymbol{d}_n) \neq \emptyset$

Por (i), $\boldsymbol{D}_n = \boldsymbol{0}$ pois $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\}$ é uma base do $\mathbb{R}^n$.

Logo, $P(\boldsymbol{D}_n, \boldsymbol{d}_n)$ é vazio se e somente se alguma coordenada de $\boldsymbol{d}_n$ é negativa

Idem para $P(\boldsymbol{A}, \boldsymbol{b})$

O algoritmo também funciona para uma base ortogonal qualquer (fazendo as projeções nas direções da base)

Projetando em um espaço menor

Seja $P(\boldsymbol{A}, \boldsymbol{b}) \subseteq \mathbb{R}^n$ um poliedro e k um inteiro positivo tal que $k + p = n$. A projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ em $\mathbb{R}^k$ é o conjunto

$$\left\{ \boldsymbol{x} \in \mathbb{R}^k \text{: existe } \boldsymbol{y} \in \mathbb{R}^p \text{ com } \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} \in P(\boldsymbol{A}, \boldsymbol{b}) \right\}$$

Para se obter tal projeção, basta considerar a projeção de $P(\boldsymbol{A}, \boldsymbol{b})$ nas direções $\{\boldsymbol{e}_n, \boldsymbol{e}_{n-1}, \ldots, \boldsymbol{e}_{n-p}\}$ e considerar o conjunto de vetores $\boldsymbol{x} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k)$ tal que $(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_k, 0, \ldots, 0)$ pertence à projeção.