Arrangements d'hyperplans : apports de la dualité et de l'optimisation

Baptiste Plaquevent-Jourdain, avec
Jean-Pierre Dussault, Université de Sherbrooke
Jean Charles Gilbert, INRIA Paris

10 Juin 2023

Outline

Plan

Définition des arrangements

- dans $\mathbb{R}^n$
- avec p hyperplans
- un hyperplan : sous-espace linéaire de dimension $n-1$ orthogonal à un vecteur v
- données : $n, p, V = [v_1 \ v_2 \ \dots \ v_p]$, on note $H_i = v_i^\perp = \{x \in \mathbb{R}^n : v_i^\top x = 0\}$

$\exists$ autre 'version' du problème : hyperplans affines $H_i = \{x \in \mathbb{R}^n : v_i^\top x = \tau_i\}$ $(\tau_i \in \mathbb{R})$

Définition des arrangements

- dans $\mathbb{R}^n$
- avec p hyperplans
- un hyperplan : sous-espace linéaire de dimension $n-1$ orthogonal à un vecteur v
- données : $n, p, V = [v_1 \ v_2 \ \dots \ v_p]$, on note $H_i = v_i^{\perp} = \{x \in \mathbb{R}^n : v_i^{\mathsf{T}} x = 0\}$

$\exists$ autre 'version' du problème : hyperplans affines $H_i = \{x \in \mathbb{R}^n : v_i^{\mathsf{T}} x = \tau_i\}$ $(\tau_i \in \mathbb{R})$

Définition des arrangements

- dans $\mathbb{R}^n$
- avec p hyperplans
- un hyperplan : sous-espace linéaire de dimension $n-1$ orthogonal à un vecteur v
- données : $n, p, V = [v_1 \ v_2 \ \dots \ v_p]$, on note $H_i = v_i^\perp = \{x \in \mathbb{R}^n : v_i^\top x = 0\}$

$\exists$ autre 'version' du problème : hyperplans affines $H_i = \{x \in \mathbb{R}^n : v_i^\top x = \tau_i\}$ $(\tau_i \in \mathbb{R})$

Définition des arrangements

- dans $\mathbb{R}^n$
- avec p hyperplans
- un hyperplan : sous-espace linéaire de dimension $n-1$ orthogonal à un vecteur v
- données : $n, p, V = [v_1 \ v_2 \ \dots \ v_p]$, on note $H_i = v_i^{\perp} = \{x \in \mathbb{R}^n : v_i^{\top} x = 0\}$

$\exists$ autre 'version' du problème : hyperplans affines
$H_i = \{x \in \mathbb{R}^n : v_i^{\top} x = \tau_i\}$ $(\tau_i \in \mathbb{R})$

Définition des arrangements

- dans $\mathbb{R}^n$
- avec p hyperplans
- un hyperplan : sous-espace linéaire de dimension $n-1$ orthogonal à un vecteur v
- données : $n, p, V = [v_1 \ v_2 \ \dots \ v_p]$, on note $H_i = v_i^\perp = \{x \in \mathbb{R}^n : v_i^\mathsf{T} x = 0\}$

$\exists$ autre 'version' du problème : hyperplans affines
$H_i = \{x \in \mathbb{R}^n : v_i^\mathsf{T} x = \tau_i\}$ $(\tau_i \in \mathbb{R})$

Vecteurs de signes et chambres

Demi-espaces d'un hyperplan

$$\mathbb{R}^n = H_i^- \cup H_i \cup H_i^+, \qquad \begin{array}{l} H_i^- = \{x \in \mathbb{R}^n : v_i^T x < 0\} \\ H_i^+ = \{x \in \mathbb{R}^n : v_i^T x > 0\} \end{array}$$

Tous les hyperplans : chaque $x \in \mathbb{R}^n$ est sur H_i, dans H_i^- ou dans H_i^+ pour chaque indice i.

Juste les "chambres" / "régions" := les $-$ et $+$ $(\mathbb{R}^n \setminus \cup H_i)$

Vecteurs de signes et chambres

Demi-espaces d'un hyperplan

$$\mathbb{R}^n = H_i^- \cup H_i \cup H_i^+, \qquad \begin{aligned} H_i^- &= \{x \in \mathbb{R}^n : v_i^\top x < 0\} \\ H_i^+ &= \{x \in \mathbb{R}^n : v_i^\top x > 0\} \end{aligned}$$

Tous les hyperplans : chaque $x \in \mathbb{R}^n$ est sur H_i, dans H_i^- ou dans H_i^+ pour chaque indice i.

Juste les "chambres" / "régions" := les $-$ et $+$ $(\mathbb{R}^n \setminus \cup H_i)$

Précision de l'objectif

Chacun des p hyperplans : H_i^- et H_i^+, donc 2^p possibilités.
On veut trouver quelles chambres sont non-vides :
$\forall\ s = (s_1, \ldots, s_p) \in \{-1, +1\}^p, \exists\ x : \forall\ i \in [1 : p], s_i v_i^\top x > 0$?

$$\begin{cases} s_1 v_1^\top x > 0 \\ \vdots \\ s_p v_p^\top x > 0 \end{cases} \Leftrightarrow \begin{cases} \min & t \\ \text{s.t.} & t \geq -1 \\ & t \geq -s_i v_i^\top x \end{cases} \qquad \text{optimisation linéaire}$$

L'ensemble des s dont le système est réalisable $:= \mathcal{S}$.

"Force brute" : vérifier 2^p systèmes. Facile de vérifier 1, mais 2^p...

Précision de l'objectif

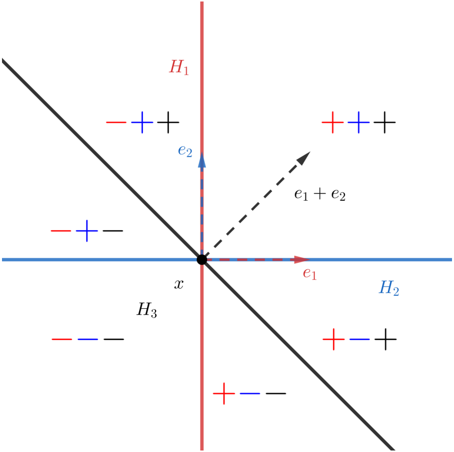
Chacun des p hyperplans : H_i^- et H_i^+, donc 2^p possibilités.
On veut trouver quelles chambres sont non-vides :
$\forall\ s = (s_1, \dots, s_p) \in \{-1, +1\}^p, \exists\ x : \forall\ i \in [1:p], s_i v_i^\top x > 0$?

$$\begin{cases} s_1 v_1^\top x > 0 \\ \vdots \\ s_p v_p^\top x > 0 \end{cases} \Leftrightarrow \begin{cases} \min & t \\ \text{s.t.} & t \geq -1 \\ & t \geq -s_i v_i^\top x \end{cases} \quad \text{optimisation linéaire}$$

L'ensemble des s dont le système est réalisable $:= \mathcal{S}$.

"Force brute" : vérifier 2^p systèmes. Facile de vérifier 1, mais 2^p...

Illustration



$H_1 = e_1^\perp, H_2 = e_2^\perp, H_3 = (e_1 + e_2)^\perp$. L'origine est x (ou l'$\cap H_i$).

Littérature

Très connu & étudié ; de façon très théorique.

Outils : fonction de Möbius, treillis (lattices), matroïdes

Des résultats stupéfiants : formule exacte, expressions très simples sous certaines hypothèses etc (Winder, [**winder-1966**])

$$
\begin{aligned}
|\{\text{chambres}\}| &= \sum_{T \subset \{H_i, i \in [1:m]\}} (-1)^{|T| - n + \dim(\bigcap H_t, t \in T)} \\
&= \sum_{\mathcal{V} \subset \{v_1, \dots, v_m\}} (-1)^{|\mathcal{V}| - \operatorname{rang}(\mathcal{V})} \\
&= \sum_{\mathcal{V} \subset \{v_1, \dots, v_m\}} (-1)^{\dim(\mathcal{N}(\mathcal{V}))}
\end{aligned}
$$

Littérature

Très connu & étudié ; de façon très théorique.

Outils : fonction de Möbius, treillis (lattices), matroïdes
Des résultats stupéfiants : formule exacte, expressions très simples sous certaines hypothèses etc (Winder, [**winder-1966**])

$$
\begin{aligned}
|\{\text{chambres}\}| &= \sum_{T \subset \{H_i, i \in [1:m]\}} (-1)^{|T| - n + \dim(\bigcap H_t, t \in T)} \\
&= \sum_{\mathcal{V} \subset \{v_1, \ldots, v_m\}} (-1)^{|\mathcal{V}| - \operatorname{rang}(\mathcal{V})} \\
&= \sum_{\mathcal{V} \subset \{v_1, \ldots, v_m\}} (-1)^{\dim(\mathcal{N}(\mathcal{V}))}
\end{aligned}
$$

Plan

Schéma

L'algorithme RČ [cerny-rada-2018]

- procédé récursif : hyperplans ajoutés un par un
- arbre - deux descendants si H_{k+1} coupe la chambre en 2
- vérification faite par l'optimisation linéaire

$\leadsto$ Il est possible d'apporter des améliorations 'générales'.
Ou d'utiliser les matroïdes !

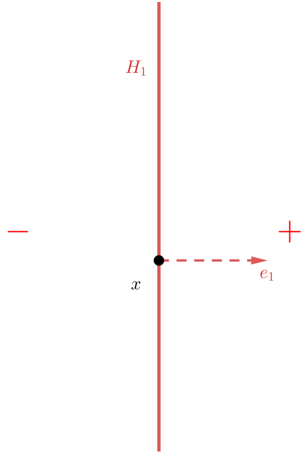
Schéma

L'algorithme RČ [cerny-rada-2018]

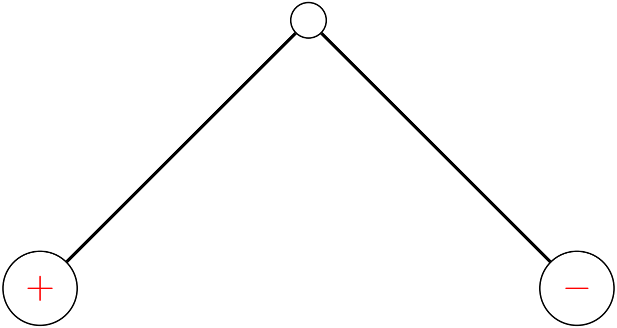
- procédé récursif : hyperplans ajoutés un par un
- arbre - deux descendants si H_{k+1} coupe la chambre en 2
- vérification faite par l'optimisation linéaire

$\rightsquigarrow$ Il est possible d'apporter des améliorations 'générales'.
Ou d'utiliser les matroïdes !

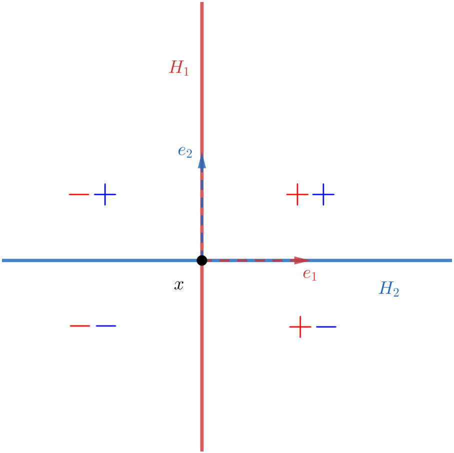
Example avec les chambres et l'arbre



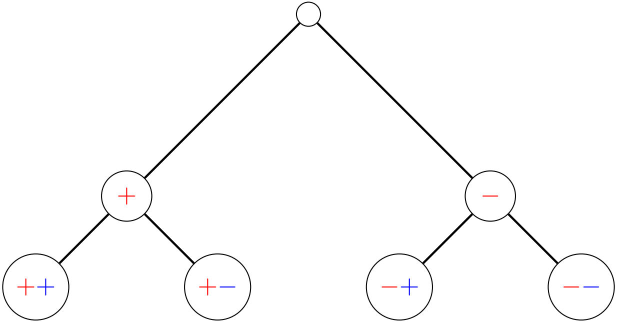
Example avec les chambres et l'arbre



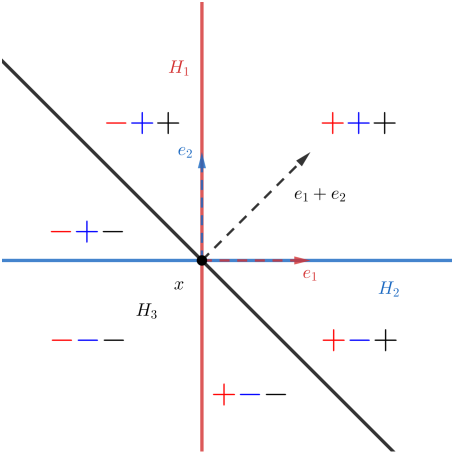
Example avec les chambres et l'arbre



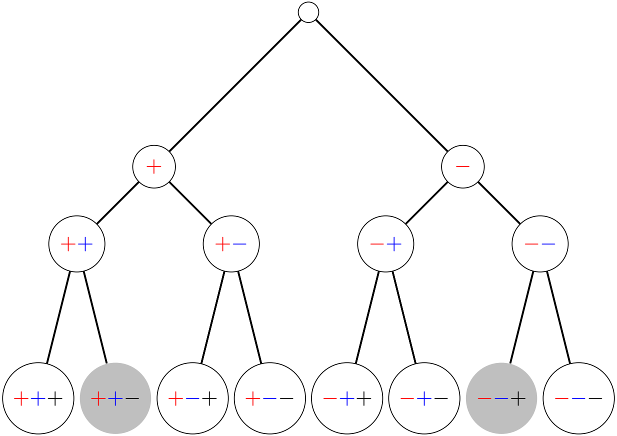
Example avec les chambres et l'arbre



Example avec les chambres et l'arbre

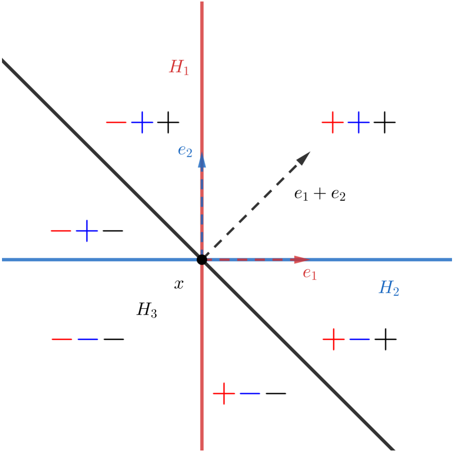


Example avec les chambres et l'arbre



Plan

Principe



$++-$ (,$--+$) correspond à un POL non-réalisable. $+$ à droite de H_1, $+$ en haut de H_2, $-$ en bas à gauche de H_3 : impossible de trouver un point.

L'alternative de Gordan

L'astuce de l'alternative

$M \in \mathbb{R}^{p \times n}$, exactement une des deux affirmations est vraie :

$$\begin{cases} \exists\ x \in \mathbb{R}^n : Mx > 0_{\mathbb{R}^p} \\ \exists\ \gamma \in \mathbb{R}^p_+ : M^\mathsf{T}\gamma = 0 \end{cases} \tag{1}$$

si $M = \mathrm{diag}(s)V^\mathsf{T} \to Mx = (s_1 v_1^\mathsf{T} x; \ldots; s_p v_p^\mathsf{T} x)$, c'est:
"il existe un point dans la chambre définie par s **ou** un $\gamma \in \mathbb{R}^p_+$"

En optimisation : "dualité" $\simeq$ autre approche/vision

L'alternative de Gordan

L'astuce de l'alternative

$M \in \mathbb{R}^{p\times n}$, exactement une des deux affirmations est vraie :

$$\begin{cases} \exists\, x \in \mathbb{R}^n : Mx > 0_{\mathbb{R}^p} \\ \exists\, \gamma \in \mathbb{R}_+^p : M^\mathsf{T}\gamma = 0 \end{cases} \tag{1}$$

si $M = \mathrm{diag}(s)V^\mathsf{T} \rightarrow Mx = (s_1 v_1^\mathsf{T} x; \ldots ; s_p v_p^\mathsf{T} x)$, c'est:
"il existe un point dans la chambre définie par s **ou** un $\gamma \in \mathbb{R}_+^p$"

En optimisation : "dualité" $\simeq$ autre approche/vision

Algorithme "dual"

Principe

Détecter les incompatibilités := les γ via l'alternative de Gordan
Arrêter une branche de l'arbre si incompatibilité

Une incompatibilité = système non-réalisable
"de taille minimale"

C'est (une utilité possible) des matroïdes !

Algorithme "dual"

Principe

Détecter les incompatibilités := les γ via l'alternative de Gordan
Arrêter une branche de l'arbre si incompatibilité

Une incompatibilité = système non-réalisable
"de taille minimale"

C'est (une utilité possible) des matroïdes !

Algorithme "dual"

Principe

Détecter les incompatibilités := les γ via l'alternative de Gordan
Arrêter une branche de l'arbre si incompatibilité

Une incompatibilité = système non-réalisable
"de taille minimale"

C'est (une utilité possible) des matroïdes !

Matroïdes linéaires/vectoriels

Définition : circuits de matroïde

Pour $V \in \mathbb{R}^{n \times p}$, $V = [v_1 \ \dots \ v_p]$, on cherche les circuits de V := $J \subset [1:p]$, $V_{:,J} = [v_j]_{j \in J}$ a un noyau de dimension $= 1$ et $\forall \ J_0 \subsetneq J, \{v_j\}_{j \in J_0}$ indépendants

$\rightarrow$ trouver tous les J:

$$J \text{ circuit} \Rightarrow \exists \eta \in \mathbb{R}^J_*, V_{:,J}\eta = 0 \Leftrightarrow [V_{:,J} \overbrace{\operatorname{sign}(\eta)][\operatorname{sign}(\eta)}^{=Id} \eta] = 0$$

$M_{(J)} = [V_{:,J}\operatorname{sign}(\eta)]$ et $\gamma_{(J)} = [\operatorname{sign}(\eta)\eta] \geq 0$ vérifient l'alternative la 'chambre' de taille $|J|$ définie par $\operatorname{sign}(\eta)$ est vide !

Matroïdes linéaires/vectoriels

Définition : circuits de matroïde

Pour $V \in \mathbb{R}^{n\times p}$, $V = [v_1 \ \dots \ v_p]$, on cherche les circuits de V $:= J \subset [1:p]$, $V_{:,J} = [v_j]_{j\in J}$ a un noyau de dimension $= 1$ et $\forall\ J_0 \subsetneq J, \{v_j\}_{j\in J_0}$ indépendants

$\rightarrow$ trouver tous les J:

$$J \text{ circuit} \Rightarrow \exists \eta \in \mathbb{R}^J_*, V_{:,J}\eta = 0 \Leftrightarrow [V_{:,J} \overbrace{\mathrm{sign}(\eta)][\mathrm{sign}(\eta)}^{=Id} \eta] = 0$$

$M_{(J)} = [V_{:,J}\mathrm{sign}(\eta)]$ et $\gamma_{(J)} = [\mathrm{sign}(\eta)\eta] \geq 0$ vérifient l'alternative la 'chambre' de taille $|J|$ définie par $\mathrm{sign}(\eta)$ est vide !

Matroïdes linéaires/vectoriels

Définition : circuits de matroïde

Pour $V \in \mathbb{R}^{n\times p}$, $V = [v_1 \;\; \ldots \;\; v_p]$, on cherche les circuits de V $:= J \subset [1:p]$, $V_{:,J} = [v_j]_{j\in J}$ a un noyau de dimension $= 1$ et $\forall\; J_0 \subsetneq J, \{v_j\}_{j\in J_0}$ indépendants

$\rightarrow$ trouver tous les J:

$$J \text{ circuit} \Rightarrow \exists \eta \in \mathbb{R}^J_*, V_{:,J}\eta = 0 \Leftrightarrow [V_{:,J}\overbrace{\mathrm{sign}(\eta)][\mathrm{sign}(\eta)}^{=Id}\eta] = 0$$

$M_{(J)} = [V_{:,J}\mathrm{sign}(\eta)]$ et $\gamma_{(J)} = [\mathrm{sign}(\eta)\eta] \geq 0$ vérifient l'alternative la 'chambre' de taille $|J|$ définie par $\mathrm{sign}(\eta)$ est vide !

Résumé

Algorithmes

- arbre normal ([**cerny-rada-2018**])
- arbre dual (matroïdes)
- arbre avec les matroïdes (mais bcp d'optimisation)

Des améliorations générales pour les algorithmes 'normaux'.
Ou des versions avec matroïdes qui améliorent beaucoup.

Résumé

Algorithmes

- arbre normal ([**cerny-rada-2018**])
- arbre dual (matroïdes)
- arbre avec les matroïdes (mais bcp d'optimisation)

Des améliorations générales pour les algorithmes 'normaux'.
Ou des versions avec matroïdes qui améliorent beaucoup.

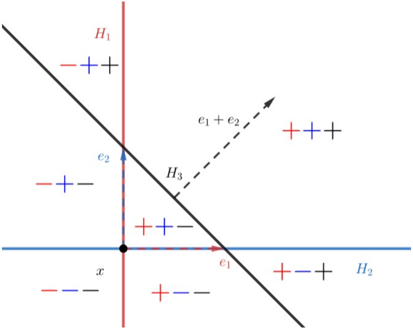
Comparaisons numériques

	temps CPU (en s)						
	original	Arbre+M-1		Arbre+M-2		Juste matroïdes	
Problème	code	Temps	Ratio	Temps	Ratio	Temps	Ratio
`rand-4-8-2`	1.06	0.10	10.75	0.02	48.44	0.03	36.67
`rand-7-9-4`	1.13	0.45	2.51	0.29	3.95	0.02	68.67
`rand-7-13-5`	11.06	4.29	2.58	2.94	3.76	0.25	44.60
`rand-8-15-7`	64.79	29.53	2.19	27.59	2.35	4.54	14.29
`rand-9-16-8`	157.05	78.01	2.01	81.61	1.92	18.87	8.32
`rand-10-17-9`	352.42	196.09	1.80	213.48	1.65	70.19	5.02
`srand-8-20-4`	874.01	323.56	2.70	649.61	1.35	705.36	1.24
`rc-2d-20-6`	12.68	0.35	36.06	0.26	48.78	0.26	49.63
`rc-2d-20-7`	23.01	0.56	40.87	0.53	43.06	0.45	51.50
`rc-perm-6`	62.89	0.84	74.44	2.33	27.03	2.46	25.61
`rc-perm-8`	6589.31	85.70	76.89	1599.53	4.12	5290.13	1.25
`rc-ratio-20-5-7`	91.57	27.43	3.34	29.70	3.08	20.54	4.46
`rc-ratio-20-5-9`	88.24	25.21	3.50	27.54	3.20	17.75	4.97
`rc-ratio-20-7-7`	581.28	241.24	2.41	506.67	1.15	447.83	1.30
`rc-ratio-20-7-9`	460.64	162.95	2.83	315.67	1.46	234.72	1.96
Moyenne (totale)			16.60		13.90		30.31
Médiane (totale)			3.24		4.12		27.80

Plan

Notations

$$H_i = \{x \in \mathbb{R}^n : v_i^\mathsf{T} x = 0\} \quad \rightarrow \quad H_i = \{x \in \mathbb{R}^n : v_i^\mathsf{T} x = \tau_i\}$$
$$\text{linéaire/homogène} \quad \rightarrow \quad \text{affine/non} - \text{homogène}$$



Une chambre auparavant vide, $++-$, est maintenant non-vide.

Gordan et Motzkin

$$\underbrace{\begin{cases} s_1 v_1^\mathsf{T} x > 0 \\ \vdots \\ s_p v_p^\mathsf{T} x > 0 \end{cases}}_{\text{Gordan}} \rightsquigarrow \underbrace{\begin{cases} s_1 v_1^\mathsf{T} x > s_1 \tau_1 \\ \vdots \\ s_p v_p^\mathsf{T} x > s_p \tau_p \end{cases}}_{\text{Motzkin}}$$

L'autre alternative (Motzkin)

$M \in \mathbb{R}^{p \times n} \backslash \{0\}$, $m \in \mathbb{R}^p$, exactement une des affirmations est vraie

$$\begin{cases} \exists\, x \in \mathbb{R}^n : Mx > m \\ \exists\, (\gamma, \gamma_0) \in (\mathbb{R}^p_+ \times \mathbb{R}^-) \backslash \{0\} : M^\mathsf{T} \gamma = 0,\, m^\mathsf{T} \gamma + \gamma_0 = 0 \end{cases} \tag{2}$$

Gordan et Motzkin

$$\underbrace{\begin{cases} s_1 v_1^\mathsf{T} x > 0 \\ \vdots \\ s_p v_p^\mathsf{T} x > 0 \end{cases}}_{\text{Gordan}} \rightsquigarrow \underbrace{\begin{cases} s_1 v_1^\mathsf{T} x > s_1 \tau_1 \\ \vdots \\ s_p v_p^\mathsf{T} x > s_p \tau_p \end{cases}}_{\text{Motzkin}}$$

L'autre alternative (Motzkin)

$M \in \mathbb{R}^{p \times n} \setminus \{0\}, m \in \mathbb{R}^p$, exactement une des affirmations est vraie

$$\begin{cases} \exists\, x \in \mathbb{R}^n : Mx > m \\ \exists\, (\gamma, \gamma_0) \in (\mathbb{R}^p_+ \times \mathbb{R}^-) \setminus \{0\} : M^\mathsf{T} \gamma = 0, m^\mathsf{T} \gamma + \gamma_0 = 0 \end{cases} \tag{2}$$

Résumé

- algorithme initial avec l'arbre & améliorations
- algorithme avec 100% matroïdes
- applicables aussi dans le cas affine, légères adaptations

L'option mixte

Regarder l'arrangement dans $\mathbb{R}^{n+1}$, $\tau_i \leftrightarrow$ nouvelle dimension.
L'idéal : résoudre le cas affine par algorithme pour le cas linéaire.

Résumé

- algorithme initial avec l'arbre & améliorations
- algorithme avec 100% matroïdes
- applicables aussi dans le cas affine, légères adaptations

L'option mixte

Regarder l'arrangement dans $\mathbb{R}^{n+1}$, $\tau_i \leftrightarrow$ nouvelle dimension.
L'idéal : résoudre le cas affine par algorithme pour le cas linéaire.

L'approche mixte - notations

$$V = [v_1 \dots v_p] \quad \widetilde{V} = \begin{bmatrix} V \\ T \end{bmatrix} = [\widetilde{v}_1 \;\; \dots \;\; \widetilde{v}_p] = \begin{bmatrix} v_1 & \dots & v_p \\ \tau_1 & \dots & \tau_p \end{bmatrix}$$
$$T = [\tau_1 \dots \tau_p]$$

- $\mathcal{S}^H(V) = \{s : \exists\, x, \mathrm{diag}(s) V^\top x > 0\}$ cas linéaire;
- $\mathcal{S}^{nH}(\widetilde{V}) = \{s : \exists\, x, \mathrm{diag}(s) V^\top x > s \cdot T^\top\}$ for $\widetilde{V} := (V; T)$, cas affine;
- $\mathcal{S}^H(\widetilde{V}) = \{s : \exists\, \widetilde{x} \in \mathbb{R}^{\mathbf{n+1}}, \mathrm{diag}(s) \widetilde{V}^\top \widetilde{x} > 0\}$, cas mixte := regarder $\widetilde{V}$ comme cas linéaire dans $\mathbb{R}^{(\mathbf{n+1}) \times p}$.

L'approche mixte - notations

$$V = [v_1 \dots v_p] \quad T = [\tau_1 \dots \tau_p] \quad \widetilde{V} = \begin{bmatrix} V \\ T \end{bmatrix} = [\widetilde{v}_1 \ \dots \ \widetilde{v}_p] = \begin{bmatrix} v_1 & \dots & v_p \\ \tau_1 & \dots & \tau_p \end{bmatrix}$$

- $\mathcal{S}^H(V) = \{s : \exists\ x, \mathrm{diag}(s) V^\top x > 0\}$ cas linéaire;
- $\mathcal{S}^{nH}(\widetilde{V}) = \{s : \exists\ x, \mathrm{diag}(s) V^\top x > s \cdot T^\top\}$ for $\widetilde{V} := (V; T)$, cas affine;
- $\mathcal{S}^H(\widetilde{V}) = \{s : \exists\ \widetilde{x} \in \mathbb{R}^{\mathbf{n+1}}, \mathrm{diag}(s) \widetilde{V}^\top \widetilde{x} > 0\}$, cas mixte := regarder $\widetilde{V}$ comme cas linéaire dans $\mathbb{R}^{(\mathbf{n+1}) \times p}$.

L'approche mixte - notations

$$V = [v_1 \dots v_p] \quad T = [\tau_1 \dots \tau_p] \quad \widetilde{V} = \begin{bmatrix} V \\ T \end{bmatrix} = [\widetilde{v}_1 \;\; \dots \;\; \widetilde{v}_p] = \begin{bmatrix} v_1 & \dots & v_p \\ \tau_1 & \dots & \tau_p \end{bmatrix}$$

- $\mathcal{S}^H(V) = \{s : \exists\, x, \mathrm{diag}(s) V^\top x > 0\}$ cas linéaire;
- $\mathcal{S}^{nH}(\widetilde{V}) = \{s : \exists\, x, \mathrm{diag}(s) V^\top x > s \cdot T^\top\}$ for $\widetilde{V} := (V; T)$, cas affine;
- $\mathcal{S}^H(\widetilde{V}) = \{s : \exists\, \widetilde{x} \in \mathbb{R}^{\mathbf{n+1}}, \mathrm{diag}(s) \widetilde{V}^\top \widetilde{x} > 0\}$, cas mixte := regarder $\widetilde{V}$ comme cas linéaire dans $\mathbb{R}^{(\mathbf{n+1}) \times p}$.

L'approche mixte - notations

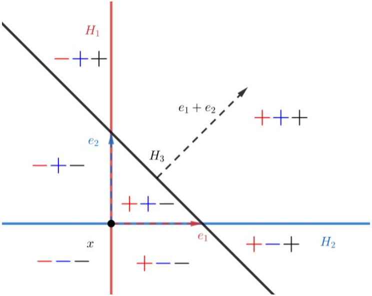
$$V = [v_1 \dots v_p] \quad \widetilde{V} = \begin{bmatrix} V \\ T \end{bmatrix} = [\widetilde{v}_1 \;\; \dots \;\; \widetilde{v}_p] = \begin{bmatrix} v_1 & \dots & v_p \\ \tau_1 & \dots & \tau_p \end{bmatrix}$$
$$T = [\tau_1 \dots \tau_p]$$

- $\mathcal{S}^H(V) = \{s : \exists\ x, \mathrm{diag}(s)V^\mathsf{T}x > 0\}$ cas linéaire;
- $\mathcal{S}^{nH}(\widetilde{V}) = \{s : \exists\ x, \mathrm{diag}(s)V^\mathsf{T}x > s \cdot T^\mathsf{T}\}$ for $\widetilde{V} := (V; T)$, cas affine;
- $\mathcal{S}^H(\widetilde{V}) = \{s : \exists\ \widetilde{x} \in \mathbb{R}^{\mathbf{n+1}}, \mathrm{diag}(s)\widetilde{V}^\mathsf{T}\widetilde{x} > 0\}$, cas mixte := regarder $\widetilde{V}$ comme cas linéaire dans $\mathbb{R}^{(\mathbf{n+1})\times p}$.

L'approche mixte - fin des notations

Partie symétrique et asymétrique d'un cas affine

$$\begin{cases} \mathcal{S}^{nH}_{sym}(\widetilde{V}) & := & \mathcal{S}^{nH}(\widetilde{V}) \cap [-\mathcal{S}^{nH}(\widetilde{V})] \\ \mathcal{S}^{nH}_{asym}(\widetilde{V}) & := & \mathcal{S}^{nH}(\widetilde{V}) \setminus \mathcal{S}^{nH}_{sym}(\widetilde{V}) \end{cases}$$



La nouvelle chambre $++-$ est asymétrique, les autres sont symétriques.

L'approche mixte - propriétés

$$\mathcal{S}^H(V) \subseteq \mathcal{S}^H(\widetilde{V}) \tag{3}$$

$$\mathcal{S}^H(V)^c \supseteq \mathcal{S}^H(\widetilde{V})^c \tag{4}$$

$$\mathcal{S}^{nH}_{sym}(\widetilde{V}) = \mathcal{S}^H(\widetilde{V}) \cap \mathcal{S}^H(V) = \mathcal{S}^H(V) \tag{5}$$

$$\mathcal{S}^{nH}_{asym}(\widetilde{V}) \cup [-\mathcal{S}^{nH}_{asym}(\widetilde{V})] = \mathcal{S}^H(\widetilde{V}) \backslash \mathcal{S}^H(V) \tag{6}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{sym} = \mathcal{S}^H(\widetilde{V})^c \cap \mathcal{S}^H(V)^c = \mathcal{S}^H(\widetilde{V})^c \tag{7}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} \cup -[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} = \mathcal{S}^H(V)^c \backslash \mathcal{S}^H(\widetilde{V})^c \tag{8}$$

L'approche mixte - propriétés

$$\mathcal{S}^H(V) \subseteq \mathcal{S}^H(\widetilde{V}) \tag{3}$$

$$\mathcal{S}^H(V)^c \supseteq \mathcal{S}^H(\widetilde{V})^c \tag{4}$$

$$\mathcal{S}^{nH}_{sym}(\widetilde{V}) = \mathcal{S}^H(\widetilde{V}) \cap \mathcal{S}^H(V) = \mathcal{S}^H(V) \tag{5}$$

$$\mathcal{S}^{nH}_{asym}(\widetilde{V}) \cup [-\mathcal{S}^{nH}_{asym}(\widetilde{V})] = \mathcal{S}^H(\widetilde{V}) \backslash \mathcal{S}^H(V) \tag{6}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{sym} = \mathcal{S}^H(\widetilde{V})^c \cap \mathcal{S}^H(V)^c = \mathcal{S}^H(\widetilde{V})^c \tag{7}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} \cup -[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} = \mathcal{S}^H(V)^c \backslash \mathcal{S}^H(\widetilde{V})^c \tag{8}$$

L'approche mixte - propriétés

$$\mathcal{S}^H(V) \subseteq \mathcal{S}^H(\widetilde{V}) \tag{3}$$

$$\mathcal{S}^H(V)^c \supseteq \mathcal{S}^H(\widetilde{V})^c \tag{4}$$

$$\mathcal{S}^{nH}_{sym}(\widetilde{V}) = \mathcal{S}^H(\widetilde{V}) \cap \mathcal{S}^H(V) = \mathcal{S}^H(V) \tag{5}$$

$$\mathcal{S}^{nH}_{asym}(\widetilde{V}) \cup [-\mathcal{S}^{nH}_{asym}(\widetilde{V})] = \mathcal{S}^H(\widetilde{V}) \backslash \mathcal{S}^H(V) \tag{6}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{sym} = \mathcal{S}^H(\widetilde{V})^c \cap \mathcal{S}^H(V)^c = \mathcal{S}^H(\widetilde{V})^c \tag{7}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} \cup -[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} = \mathcal{S}^H(V)^c \backslash \mathcal{S}^H(\widetilde{V})^c \tag{8}$$

L'approche mixte - propriétés

$$\mathcal{S}^H(V) \subseteq \mathcal{S}^H(\widetilde{V}) \tag{3}$$

$$\mathcal{S}^H(V)^c \supseteq \mathcal{S}^H(\widetilde{V})^c \tag{4}$$

$$\mathcal{S}^{nH}_{sym}(\widetilde{V}) = \mathcal{S}^H(\widetilde{V}) \cap \mathcal{S}^H(V) = \mathcal{S}^H(V) \tag{5}$$

$$\mathcal{S}^{nH}_{asym}(\widetilde{V}) \cup [-\mathcal{S}^{nH}_{asym}(\widetilde{V})] = \mathcal{S}^H(\widetilde{V}) \backslash \mathcal{S}^H(V) \tag{6}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{sym} = \mathcal{S}^H(\widetilde{V})^c \cap \mathcal{S}^H(V)^c = \mathcal{S}^H(\widetilde{V})^c \tag{7}$$

$$[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} \cup -[\mathcal{S}^{nH}(\widetilde{V})^c]_{asym} = \mathcal{S}^H(V)^c \backslash \mathcal{S}^H(\widetilde{V})^c \tag{8}$$

Arguments et intérêt

Preuves : les définitions & basculer entre Gordan et Motzkin

$$\mathcal{S}^{nH}_{sym}(\widetilde{V}) = \mathcal{S}^{H}(\widetilde{V}) \cap \mathcal{S}^{H}(V) = \mathcal{S}^{H}(V)$$

$$\mathcal{S}^{nH}_{asym}(\widetilde{V}) \cup [-\mathcal{S}^{nH}_{asym}(\widetilde{V})] = \mathcal{S}^{H}(\widetilde{V}) \backslash \mathcal{S}^{H}(V)$$

Droite : uniquement des cas linéaires
Gauche : le cas affine recherché

Détail restant : distinguer les bons s des $-s$.
(Ça marche bien.)

Arguments et intérêt

Preuves : les définitions & basculer entre Gordan et Motzkin

$$\mathcal{S}^{nH}_{sym}(\widetilde{V}) = \mathcal{S}^{H}(\widetilde{V}) \cap \mathcal{S}^{H}(V) = \mathcal{S}^{H}(V)$$

$$\mathcal{S}^{nH}_{asym}(\widetilde{V}) \cup [-\mathcal{S}^{nH}_{asym}(\widetilde{V})] = \mathcal{S}^{H}(\widetilde{V}) \backslash \mathcal{S}^{H}(V)$$

Droite : uniquement des cas linéaires
Gauche : le cas affine recherché

Détail restant : distinguer les bons s des $-s$.
(Ça marche bien.)

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.
Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?
$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.
Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?
$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.
Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?
$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.
Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?
$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.
Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?
$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.

Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?

$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.
Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?
$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Conclusion

Algorithmes

- algorithme de l'arbre (cas affine/linéaire)
- améliorations structurelles
- améliorations fondamentales : matroïdes
- adaptations des améliorations au cas affine
- algorithme 'mixte' pour le cas affine

Code pour les cas affine/mixte en développement.
Ouverture : si on veut "tout" l'arrangement ($\supsetneq$ chambres) ?
$\rightsquigarrow$ remplacer $\{-,+\}$ par $\{-,0,+\}$ (mais 3^p...) Merci pour votre attention ! Des questions ?

Bibliographic elements I