

Quelques applications de la programmation semidéfinie.

Alexandre d'Aspremont

CNRS & Centre de Mathématiques Appliquées,
École Polytechnique, 91128 Palaiseau, France.

L'optimisation a subi deux mutations profondes ces vingt dernières années. La première est liée aux travaux de [Nesterov and Nemirovskii \[1994\]](#), qui ont radicalement transformé notre compréhension des questions de complexité, montrant notamment que la dichotomie classique entre problèmes linéaires et non-linéaires n'avait en fait que très peu de liens avec la véritable complexité algorithmique de ces problèmes. Ces travaux ont confirmé de manière explicite et constructive que la convexité est bien la frontière naturelle entre problèmes simples (solubles en temps polynomial) et problèmes complexes (potentiellement combinatoires et exponentiels). Les programmes linéaires avaient catalysé depuis longtemps une activité de recherche intense, débouchant sur une longue série d'applications en recherche opérationnelle, réseaux, statistiques, finance, électronique, signal, etc., et les travaux de [Nesterov and Nemirovskii \[1994\]](#) ont généralisé ces résultats à une classe beaucoup plus large de problèmes.

Un programme linéaire s'écrit

$$\begin{aligned} & \text{minimize} && c^T x \\ & \text{subject to} && Ax = b \\ & && x \succeq 0, \end{aligned} \tag{1}$$

dans la variable $x \in \mathbb{R}^n$, et les paramètres $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$ et $c \in \mathbb{R}^n$. Initialement, ces programmes étaient résolus en utilisant une variante de l'algorithme du simplexe [[Dantzig, 1963](#)]. Ces méthodes de nature combinatoire sont très efficaces en pratique mais leur complexité est exponentielle dans le pire des cas. Les algorithmes dérivés de la méthode de l'ellipsoïde [[Nemirovskii and Yudin, 1979](#)], exploitant la convexité du problème, furent utilisés par [[Khachiyan, 1979](#)] pour montrer que les programmes convexes comme le programme linéaire en (1) pouvaient toujours être résolus en temps polynomial, mais il a fallu attendre les travaux de [[Karmarkar, 1984](#)] pour voir apparaître des algorithmes à la fois polynomiaux et efficaces en pratique. Ces algorithmes étaient basés sur un schéma du type point intérieur (l'algorithme converge vers la solution en suivant un chemin à l'intérieur de l'ensemble faisable), complètement différent du procédé combinatoire utilisé par le simplexe (où l'algorithme progresse d'un vertex à un autre). Les résultats de [[Nesterov and Nemirovskii, 1994](#)] ont montré que les méthodes de point intérieur utilisées pour la programmation linéaire pouvaient être directement généralisées à la résolution de programmes convexes beaucoup plus génériques. Parmi ces nouveaux problèmes dont la résolution numérique était devenue soudainement réaliste, les *programmes semidéfinis* généralisent la programmation linéaire au cône des matrices semidéfinies positives. Un programme semidéfini (SDP) s'écrit

$$\begin{aligned} & \text{minimize} && \text{Tr}(CX) \\ & \text{subject to} && \text{Tr}(A_j X) = b_j, \quad j = 1, \dots, m \\ & && X \succeq 0, \end{aligned} \tag{2}$$

dans la variable $X \in \mathbf{S}_n$, l'espace des matrices symétriques de dimension n , avec paramètres $C \in \mathbf{S}_n$, $A_j \in \mathbf{S}_n$ et $b_j \in \mathbb{R}$ pour $j = 1, \dots, m$, où $X \succeq 0$ signifie que X est semidéfinie positive (e.g. une matrice

de covariance). De nombreuses librairies numériques basées sur des méthodes de point intérieur résolvent des problèmes de taille moyenne (n de l'ordre de mille) de manière fiable et efficace, et la résolution de ces problèmes qui semblait extrêmement délicate il y a une vingtaine d'années est maintenant une question de routine. Ces progrès ont permis la découverte et le développement de nombreuses applications de la programmation semidéfinie, avec une activité très similaire à celle qui avait suivi la découverte de l'algorithme du simplexe pour la programmation linéaire. Pour ne citer que quelques exemples, les SDP ont été utilisés pour établir la stabilité de systèmes de contrôle, pour résoudre des problèmes de filtrage collaboratif (e.g. faire des recommandations de films personnalisées dans le cas du concours NETFLIX), pour approximer la solution de problèmes combinatoires (e.g. MaxCut), pour optimiser la vitesse de mélange d'une chaîne de Markov sur un réseau, pour extraire la structure de dépendance de données multivariées ou enfin pour produire des noyaux optimaux en classification. Malheureusement, la performance des méthodes de point intérieur se dégrade rapidement pour les problèmes de plus grande taille, les contraintes CPU et mémoire rendant en particulier impossible la formation et le stockage du système de Newton à tel point que résoudre ne fusse qu'une seule itération devient rapidement hors de portée lorsque la dimension n augmente. Dans cette optique, les efforts de recherche actuels ont pour objectif de dépasser ces limites strictes sur la taille des programmes semidéfinis qui peuvent être résolus efficacement, tout en obtenant une fiabilité algorithmique comparable à celle des méthodes de point intérieur.

Le deuxième développement majeur en optimisation a été purement exogène : la puissance de calcul des ordinateurs personnels a augmenté de plusieurs ordres de grandeur ces trois dernières décennies, avec un impact direct en optimisation. Il y a un peu plus d'une dizaine d'années, seuls des algorithmes très rudimentaires pouvaient être utilisés dans la résolution de problèmes de taille réaliste. A l'inverse, des algorithmes plus sophistiqués, mieux compris sur le plan théorique et beaucoup plus efficaces dans certains scénarios, ne pouvaient être utilisés que pour résoudre des problèmes de taille plutôt modeste (la programmation linéaire étant évidemment une exception importante). Ces restrictions ont aujourd'hui commencé à s'effacer, ce qui signifie que certains domaines comme l'apprentissage et les statistiques font maintenant une utilisation intensive de l'optimisation convexe, allant très au delà des quelques méthodes ad-hoc qui servaient jusqu'à présent. Cependant, alors que l'optimisation et les statistiques ont trouvé un champs d'expérimentation commun, ces nouvelles applications requièrent des algorithmes aux caractéristiques radicalement différentes (e.g. faible précision, mais données de très grande taille), comparées aux applications plus classiques en ingénierie. Au delà de ces questions algorithmiques, les questions de complexité ont maintenant gagné une place importante en statistiques, mais le problème de réconcilier les concepts de complexité algorithmique avec ceux utilisés pour décrire la performance statistique reste ouvert. De nombreuses applications montrent par exemple qu'il y a un compromis direct entre le coût numérique et le taux de convergence de certains estimateurs, mais il n'y a jusqu'à présent que très peu d'indices théoriques pour justifier ces arbitrages de manière explicite.

1 Applications

La programmation semidéfinie a trouvé de nombreuses applications en ingénierie, statistiques, apprentissage, graphes, etc. Pour ne citer que quelques exemples, les SDP apparaissent naturellement en contrôle (stabilité), apprentissage (filtrage collaboratif, etc.), analyse des séries temporelles (complétion de covariances), structure (stabilité, oscillations), graphes (mélange de chaînes de Markov), etc. Alors que reconnaître un programme linéaire est un exercice presque immédiat, la programmation semidéfinie a un pouvoir d'expression beaucoup plus fort et les SDP apparaissent souvent de manière beaucoup plus inattendue. Quelques unes de ces applications "exotiques" sont détaillées dans ce qui suit. En particulier,

les applications présentées à partir de la section 1.7 sont plus spécialisées et liées aux travaux récents de l'auteur.

1.1 Optimisation de valeurs propres

Une des applications les plus naturelles de la programmation semidéfinie est la résolution de problèmes convexes sur les valeurs propres. En particulier, un SDP contenant une contrainte sur la trace de la variable X , qui s'écrit

$$\begin{aligned} & \text{minimize} && \mathbf{Tr}(CX) \\ & \text{subject to} && \mathbf{Tr}(A_i X) = b_i, \quad i = 1, \dots, m \\ & && \mathbf{Tr}(X) = 1 \\ & && X \succeq 0, \end{aligned}$$

a pour dual

$$\min_y \lambda_{\max} \left(C - \sum_{i=1}^m y_i A_i \right) + b^T y$$

en $y \in \mathbb{R}^m$, en exploitant le fait que

$$\max_{\substack{\mathbf{Tr}(X)=1, \\ X \succeq 0}} \mathbf{Tr}(CX) = \lambda_{\max}(C).$$

Les livres de [Boyd and Vandenberghe \[2004\]](#), [Ben-Tal and Nemirovski \[2001\]](#) couvrent en détail ces problèmes et leurs généralisations.

1.2 Relaxations de problèmes combinatoires

Les SDP avec contrainte de trace apparaissent très fréquemment dans l'une des applications les plus fructueuses de la programmation semidéfinie, qui consiste à relaxer un problème quadratique non convexe en SDP après avoir effectué un simple changement de variable $X = xx^T$ où x est la variable du problème originel et X la variable de sa relaxation SDP. Cette technique due à [[Shor, 1987](#), [Lovász and Schrijver, 1991](#)], s'illustre très simplement sur un problème classique de partitionnement du type MAXCUT, qui s'écrit

$$\begin{aligned} & \max. && x^T C x \\ & \text{s.t.} && x \in \{-1, 1\}^n \end{aligned} \tag{3}$$

dans la variable binaire $x \in \{-1, 1\}^n$ avec paramètres $C \in \mathbf{S}_n$. L'objectif ici est de séparer les variables en deux ensembles $\{i : x_i = 1\}$ et $\{j : x_j = -1\}$ pour minimiser le coût total de la partition, pondéré par la matrice C . Ce problème est NP-difficile, mais peut être reformulé très simplement comme un programme quadratique non-convexe, en effet

$$x \in \{-1, 1\}^n \iff x_i^2 = 1, \quad i = 1, \dots, n$$

et le problème de partitionnement s'écrit donc

$$\begin{aligned} & \max. && x^T C x \\ & \text{s.t.} && x_i^2 = 1, \quad i = 1, \dots, n \end{aligned}$$

qui est un programme quadratique non convexe en $x \in \mathbb{R}^n$. Après un changement de variable $X = xx^T$, on obtient

$$\begin{aligned} \mathbf{Tr}(CX) &= x^T C x \\ \mathbf{diag}(X) &= \mathbf{1} \iff x_i^2 = 1, \quad i = 1, \dots, n \end{aligned}$$

et le programme quadratique peut donc s'écrire

$$\begin{aligned} \max. \quad & \mathbf{Tr}(CX) \\ \text{s.t.} \quad & \mathbf{diag}(X) = 1 \\ & X \succeq 0, \mathbf{Rank}(X) = 1 \end{aligned}$$

qui est un programme (non convexe) dans la nouvelle variable $X \in \mathbf{S}_n$. Par contre, la seule contrainte non convexe de ce dernier programme est la contrainte de rang. Si on l'ignore, on obtient une relaxation du problème originel, et donc

$$\begin{aligned} \max. \quad & x^T C x \\ \text{s.t.} \quad & x \in \{-1, 1\}^n \end{aligned}$$

est borné par

$$\begin{aligned} \max. \quad & \mathbf{Tr}(CX) \\ \text{s.t.} \quad & \mathbf{diag}(X) = 1 \\ & X \succeq 0, \end{aligned} \tag{4}$$

qui est un SDP dans la variable $X \in \mathbf{S}_n$. Cette relaxation de la contrainte de rang est difficile à contourner et on peut montrer que tous les problèmes semi-algébriques peuvent s'écrire comme des SDP avec contraintes de rang.

Un des aspects les plus marquants de cette technique est que dans certains cas, un argument probabiliste simple mais ingénieux dû à [Goemans and Williamson \[1995\]](#), [Nesterov \[1998\]](#) permet de contrôler la qualité de l'approximation. Supposons que la matrice C est semidéfinie positive. On peut utiliser la solution $X \succeq 0$ du SDP (4) pour générer des variables Gaussiennes $v \sim \mathcal{N}(0, X)$. Si l'on projette chacun de ces vecteurs v sur $\{-1, 1\}^n$ en définissant $y = \text{sign}(v)$, on peut montrer

$$\mathbf{E}[yy^T] = \frac{2}{\pi} \arcsin(X)$$

où $\arcsin(X)_{ij} = \arcsin(X_{ij})$ ici. Sachant que $X \preceq \arcsin(X)$ lorsque $X \succeq 0$, on obtient

$$\mathbf{E}[y^T C y] = \frac{2}{\pi} \mathbf{Tr}(C \arcsin(X)) \geq \frac{2}{\pi} \mathbf{Tr}(CX).$$

Chacun de ces vecteurs y est un point faisable du problème (3), ceci signifie qu'il existe au moins une solution de (3) dont l'objectif est supérieur à $\frac{2}{\pi} \mathbf{Tr}(CX)$. En conclusion, si l'on appelle SDP la solution de (4) et OPT la solution de (3), on a montré

$$\frac{2}{\pi} SDP \leq OPT \leq SDP.$$

Ceci signifie que la solution du SDP (convexe) permet de produire une solution du problème combinatoire sous-optimale par un facteur au plus $\frac{2}{\pi}$. Ce résultat a de nombreuses applications et généralisations sur les graphes, sur l'approximation de matrices [[Frieze and Kannan, 1999](#), [Alon and Naor, 2004](#), [Nemirovski, 2005](#)], etc.

1.3 Approximations ellipsoïdales

Le problème d'approximer un ensemble par une ellipsoïde \mathcal{E} de volume optimal est convexe. Etant donné un ensemble C , supposons par exemple que l'on cherche \mathcal{E} de volume minimal telle que $C \subseteq \mathcal{E}$. En paramétrant \mathcal{E} par

$$\mathcal{E} = \{v \mid \|Av + b\|_2 \leq 1\} \quad \text{with } A \succ 0$$

vol \mathcal{E} est proportionnel à $\det A^{-1}$ et le problème d'identifier l'ellipsoïde de volume minimal contenant C , dite ellipsoïde de Löwner-John, s'écrit

$$\begin{aligned} & \text{minimize (over } A, b) \quad \log \det A^{-1} \\ & \text{subject to} \quad \sup_{v \in C} \|Av + b\|_2 \leq 1. \end{aligned}$$

Ce problème est convexe, mais la contrainte n'est pas systématiquement implémentable. Le problème se simplifie lorsque $C = \{x_1, \dots, x_m\}$ est un ensemble fini, ou un polytope avec un nombre fini de sommets, il s'écrit alors

$$\begin{aligned} & \text{minimize (over } A, b) \quad \log \det A^{-1} \\ & \text{subject to} \quad \|Ax_i + b\|_2 \leq 1, \quad i = 1, \dots, m \end{aligned}$$

dans les variables $A \in \mathbf{S}_n$ et $b \in \mathbb{R}^n$, un programme semidéfini qui produit aussi l'ellipsoïde optimale pour le polyèdre $\text{Co}\{x_1, \dots, x_m\}$.

Une approche similaire permet de résoudre le problème d'identifier une ellipsoïde de volume maximal inscrite dans un ensemble convexe $C \subset \mathbb{R}^n$. L'ellipsoïde \mathcal{E} est maintenant paramétrée par

$$\mathcal{E} = \{Bu + d \mid \|u\|_2 \leq 1\} \quad \text{avec } B \succ 0$$

vol \mathcal{E} est proportionnel à $\det B$ et le problème d'identifier \mathcal{E} s'écrit

$$\begin{aligned} & \text{maximize} \quad \log \det B \\ & \text{subject to} \quad \sup_{\|u\|_2 \leq 1} I_C(Bu + d) \leq 0 \end{aligned}$$

où $I_C(x) = 0$ pour $x \in C$ et $I_C(x) = \infty$ pour $x \notin C$. Ce problème est convexe, mais évaluer la contrainte peut-être difficile. Il se simplifie dans le cas où C est un polyèdre paramétré par ses facettes $\{x \mid a_i^T x \leq b_i, i = 1, \dots, m\}$ et devient

$$\begin{aligned} & \text{maximize} \quad \log \det B \\ & \text{subject to} \quad \|Ba_i\|_2 + a_i^T d \leq b_i, \quad i = 1, \dots, m \end{aligned}$$

un problème convexe en B , en utilisant le fait que $\sup_{\|u\|_2 \leq 1} a_i^T (Bu + d) = \|Ba_i\|_2 + a_i^T d$. La qualité de l'approximation de C par \mathcal{E} peut être quantifiée. Lorsque C est convexe borné d'intérieur non vide, l'ellipsoïde de Löwner-John, contractée par un facteur n est inscrite dans C , de même, l'ellipsoïde de volume maximal dilatée par un facteur n contient C . Ces facteurs peuvent être remplacés par \sqrt{n} quand l'ensemble C est symétrique autour de son centre. Des exemples graphiques sont détaillés dans [Boyd and Vandenberghe, 2004] par exemple.

1.4 Distorsion, plongements et métriques finies

Il n'est pas possible de borner arbitrairement le rang d'une solution d'un programme semidéfini dans le cas général. Il existe par contre des bornes pour certains programmes plus structurés, en particulier si l'on autorise une légère distorsion autour des contraintes. Un exemple est donné par le résultat suivant, dû à [Ben-Tal, Nemirovski, and Roos, 2003].

Theorem 1.1. S-lemme approximé. Soit $A_1, \dots, A_N \in \mathbf{S}_n$, semidéfinies positives, $\alpha_1, \dots, \alpha_N \in \mathbb{R}$ et une matrice $X \in \mathbf{S}_n$ telle que

$$A_i, X \succeq 0, \quad \text{Tr}(A_i X) = \alpha_i, \quad i = 1, \dots, N$$

Soit $\epsilon > 0$, il existe une matrice X_0 telle que

$$\alpha_i(1 - \epsilon) \leq \text{Tr}(A_i X_0) \leq \alpha_i(1 + \epsilon) \quad \text{et} \quad \text{Rank}(X_0) \leq 8 \frac{\log 4N}{\epsilon^2}$$

La preuve fait appel à des arguments probabilistes et en particulier des résultats de concentration sur les formes quadratiques (voir aussi [Barvinok, 2002, Ben-Tal, El Ghaoui, and Nemirovski, 2009] pour plus de détails). Ceci n'est pas complètement un hasard, puisque le résultat classique de plongement ℓ_2 dû à Johnson et Lindenstrauss, lui aussi obtenu par des méthodes probabilistes, se retrouve comme un cas particulier de ce théorème. En effet, si on se donne N vecteurs $v_i \in \mathbb{R}^d$ et que l'on construit leur matrice de Gram $X \in \mathbf{S}_N$ avec

$$X \succeq 0, \quad X_{ii} - 2X_{ij} + X_{jj} = \|v_i - v_j\|_2^2, \quad i, j = 1, \dots, N,$$

les matrices $D_{ij} \in \mathbf{S}_n$ définies par

$$\mathbf{Tr}(D_{ij}X) = X_{ii} - 2X_{ij} + X_{jj}, \quad i, j = 1, \dots, N$$

satisfont $D_{ij} \succeq 0$. Soit $\epsilon > 0$, il existe donc X_0 telle que

$$m = \mathbf{Rank}(X_0) \leq 16 \frac{\log 2N}{\epsilon^2},$$

dont on peut extraire des vecteurs $u_i \in \mathbb{R}^m$ tels que

$$\|v_i - v_j\|_2^2 (1 - \epsilon) \leq \|u_i - u_j\|_2^2 \leq \|v_i - v_j\|_2^2 (1 + \epsilon).$$

ce qui correspond au résultat du lemme de plongement de Johnson et Lindenstrauss. Le problème de reconstruire une métrique euclidienne à N points, étant donné une information partielle sur les distances entre paires de points v_i , $i = 1, \dots, N$ peut aussi s'écrire comme un SDP

$$\begin{aligned} & \text{find} && D \\ & \text{subject to} && \mathbf{1}v^T + v\mathbf{1}^T - D \succeq 0 \\ & && D_{ij} = \|v_i - v_j\|_2^2, \quad (i, j) \in S \\ & && v \geq 0 \end{aligned}$$

dans les variable $D \in \mathbf{S}_n$ et $v \in \mathbb{R}^n$ pour un sous ensemble donné $S \subset [1, N]^2$.

1.5 Mélange des chaines de Markov, dépliage de variétés

D'après [Sun, Boyd, Xiao, and Diaconis, 2006], supposons que $G = (V, E)$ est un graphe non orienté avec n sommets et m arêtes. on peut définir une chaîne de Markov sur ce graphe, en écrivant $w_{ij} \geq 0$ la probabilité de transition sur $(i, j) \in V$. Partant d'une distribution $\pi(t)$ à l'instant t , son évolution est gouvernée par l'équation de la chaleur

$$d\pi(t) = -L\pi(t)dt$$

où

$$L_{ij} = \begin{cases} -w_{ij} & \text{if } i \neq j, (i, j) \in V \\ 0 & \text{if } (i, j) \notin V \\ \sum_{(i,k) \in V} w_{ik} & \text{if } i = j \end{cases}$$

est la Laplacien du graphe (V, E) et

$$\pi(t) = e^{-Lt}\pi(0).$$

La matrice $L \in \mathbf{S}_n$ vérifie $L \succeq 0$ et sa plus petite valeur propre est zéro. Le taux de mélange de la chaîne de Markov peut donc être contrôlé par la seconde valeur propre du Laplacien $\lambda_2(L)$. Comme la première valeur propre de L est zéro, associée au vecteur propre $\mathbf{1}$, on obtient

$$\lambda_2(L) \geq t \iff L(w) \succeq t(\mathbf{I} - (1/n)\mathbf{1}\mathbf{1}^T),$$

et maximiser la borne sur le taux de mélange de la chaîne de Markov s'écrit alors

$$\begin{aligned} & \text{maximize} && t \\ & \text{subject to} && L(w) \succeq t(\mathbf{I} - (1/n)\mathbf{1}\mathbf{1}^T) \\ & && \sum_{(i,j) \in V} d_{ij}^2 w_{ij} \leq 1 \\ & && w \geq 0 \end{aligned}$$

dans les poids $w \in \mathbb{R}^m$ où $d_{ij}^2 \geq 0$ sont des facteurs de normalisation, donnés. Comme $L(w)$ est une fonction affine de la variable $w \in \mathbb{R}^m$, ce problème est un programme semidéfini. On peut montrer que ses solutions ont souvent un taux de mélange supérieur aux constructions classiques comme Metropolis-Hastings [Boyd et al., 2004]. On peut également former le dual du problème de mélange. Ce dual s'écrit

$$\begin{aligned} & \text{maximize} && \text{Tr}(X(\mathbf{I} - (1/n)\mathbf{1}\mathbf{1}^T)) \\ & \text{subject to} && X_{ii} - 2X_{ij} + X_{jj} \leq d_{ij}^2, \quad (i, j) \in V \\ & && X \succeq 0, \end{aligned}$$

dans la variable $X \in \mathbf{S}_n$. Comme dans la section précédente, cette matrice s'interprète aussi comme la matrice de Gram d'un ensemble de n points $v_i \in \mathbb{R}^d$ et le dual consiste à maximiser leur variance

$$\text{Tr}(X(\mathbf{I} - (1/n)\mathbf{1}\mathbf{1}^T)) = \sum_i \|v_i\|_2^2 - \|\sum_i v_i\|_2^2$$

alors que les contraintes bornent les distances point à point

$$X_{ii} - 2X_{ij} + X_{jj} \leq d_{ij}^2 \iff \|v_i - v_j\|_2^2 \leq d_{ij}^2.$$

Le dual du problème de mélange pour une chaîne de Markov peut donc s'interpréter comme un problème de dépliage d'une variété, maximisant la variance tout en préservant la géométrie [Weinberger and Saul, 2006, Sun et al., 2006].

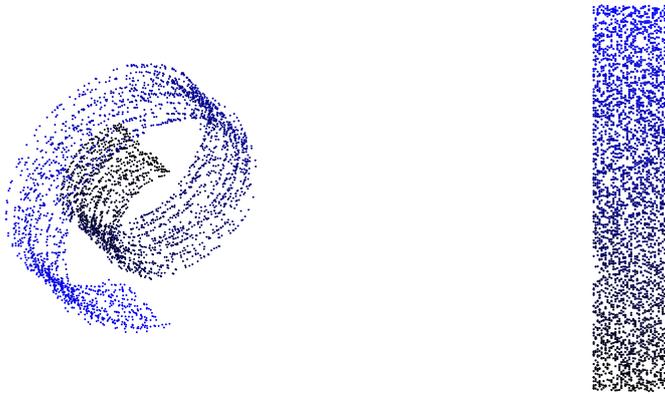


FIGURE 1: Exemple tiré de [Sun et al., 2006]. Etant donné les distances entre un point et ses k plus proches voisins (à gauche), la solution du problème de dépliage est représentée à droite.

1.6 Polynômes positifs, problèmes de moments

Le 17ème problème de Hilbert a une réponse positive en dimension un : un polynôme est positif si et seulement si il peut s'écrire comme une somme de carrés de polynômes (SOS), i.e.

$$p(x) = x^{2d} + \alpha_{2d-1}x^{2d-1} + \dots + \alpha_0 \geq 0, \text{ for all } x \iff p(x) = \sum_{i=1}^N q_i(x)^2$$

Cette condition peut s'écrire comme une inégalité matricielle linéaire [Nesterov, 2000], si on note $v(x)$ le vecteur des monômes

$$v(x) = (1, x, \dots, x^d)^T$$

on obtient

$$\sum_i \lambda_i u_i u_i^T = M \succeq 0 \iff p(x) = v(x)^T M v(x) = \sum_i \lambda_i (u_i^T v(x))^2$$

où (λ_i, u_i) sont les valeurs/vecteurs propres de M . Le cône dual du cône des coefficients de polynômes SOS est le cône des matrices de moments de mesures positives.

$$\mathbf{E}_\mu[x^i] = q_i, i = 0, \dots, d \iff \begin{pmatrix} q_0 & q_1 & \cdots & q_d \\ q_1 & q_2 & & q_{d+1} \\ \vdots & & \ddots & \vdots \\ q_d & q_{d+1} & \cdots & q_{2d} \end{pmatrix} \succeq 0$$

Les travaux de [Putinar, 1993, Lasserre, 2001, Parrilo, 2000] montrent que ces résultats peuvent se généraliser au cas multivarié et permettent notamment de formuler les problèmes d'optimisation semi-algébriques compacts comme des programmes semidéfinis (de taille potentiellement exponentielle).

1.7 Analyse en composantes principales parcimonieuse

L'analyse en composantes principales (ACP) est un outil classique pour l'analyse de données multivariées et la réduction de dimension [Jolliffe, 1986]. L'ACP explique une part importante de la dynamique de données multivariées en utilisant un nombre restreint de facteurs, tout en maximisant la variance capturée par ces facteurs. L'une des limitations principales de l'ACP est que ces facteurs sont denses, i.e. dépendent de toutes les variables originelles. Dans certaines applications de l'ACP, comme l'analyse de données génétiques par exemple, ceci rend les résultats difficiles à interpréter. Le but de l'ACP parcimonieuse est d'isoler un nombre restreint de facteurs expliquant un maximum de variance, tout en contraignant ces facteurs à être creux, i.e. à n'impliquer qu'un nombre restreint de variables. L'ACP représente les données en fonctions de facteurs obtenus par une SVD, sa version parcimonieuse s'obtient en incorporant au problème de vecteur propre une contrainte sur le support de la solution, pour résoudre

$$\begin{aligned} & \text{maximize} && x^T \Sigma x \\ & \text{subject to} && \|x\|_2 = 1 \\ & && \text{Card}(x) \leq k, \end{aligned} \tag{5}$$

dans la variable $x \in \mathbb{R}^n$, où $\Sigma \in \mathbf{S}^n$ est la matrice de covariance et $k > 0$ est un paramètre contrôlant la parcimonie de la solution. Ce problème est malheureusement non-convexe et NP-difficile. [d'Aspremont et al., 2007] détaillent une relaxation convexe qui s'écrit

$$\begin{aligned} & \text{maximize} && \text{Tr}(\Sigma X) \\ & \text{subject to} && \text{Tr}(X) = 1 \\ & && \mathbf{1}^T |X| \mathbf{1} \leq k \\ & && X \succeq 0, \end{aligned} \tag{6}$$

et devient un programme semidéfini dans la variable $X \in \mathbf{S}_n$. Un facteur creux est ensuite extrait de la solution X en calculant son vecteur propre dominant (quand la solution X est de rang un, la relaxation est exacte). Cette procédure de relaxation semidéfinie est similaire à celle utilisée pour MAXCUT un peu plus haut.

Une autre relaxation semidéfinie du problème (6) a été dérivée par [d'Aspremont et al., 2008] et produit un programme semidéfini de beaucoup plus grande dimension, en relaxant

$$\max_{\|z\| \leq 1} z^T \Sigma z - \rho \mathbf{Card}(z)$$

par les SDP primaux et duaux suivants

$$\begin{aligned} \psi(\rho) = \max. & \sum_{i=1}^n \mathbf{Tr}(P_i B_i) \\ \text{s.t.} & \mathbf{Tr}(X) = 1, X \succeq 0, X \succeq P_i \succeq 0, \end{aligned}$$

dans la variable $X \in \mathbf{S}_n$, $P_i \in \mathbf{S}_n$, et

$$\begin{aligned} \min. & \lambda_{\max}(\sum_{i=1}^n Y_i) \\ \text{s.t.} & Y_i \succeq B_i, Y_i \succeq 0, \quad i = 1, \dots, n. \end{aligned}$$

dans les variables $Y_i \in \mathbf{S}_n$.

Même si cette dernière relaxation est beaucoup plus difficile à résoudre numériquement, elle permet de tester très efficacement l'optimalité de solutions de rang un issues d'algorithmes gourmands. Les expériences numériques menées dans [d'Aspremont et al., 2008] montrent que ces tests certifient l'optimalité d'un large spectre de solutions pour des données naturelles. Une application de ces techniques à des données génétiques est illustrée dans la figure 2.

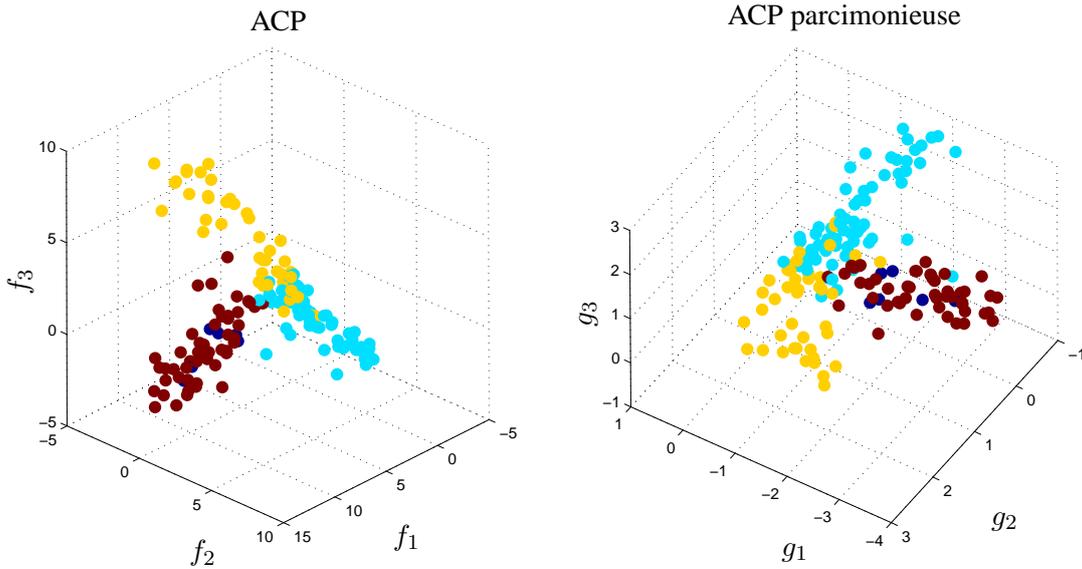


FIGURE 2: Représentation de données génétiques, pour l'ACP (gauche) et l'ACP parcimonieuse (droite), sur l'expression de 500 gènes. Les facteurs de l'ACP sont tous denses (500 gènes), alors que les facteurs g_1, g_2, g_3 produits par l'ACP parcimonieuse contiennent 6, 4 et 4 gènes respectivement. (Données : Iconix Pharmaceuticals)

1.8 Détection pour l'ACP parcimonieuse

Dans l'esprit des travaux récents de [Amini and Wainwright, 2009, Berthet and Rigollet, 2012], et prolongeant les résultats détaillés dans ce qui précède, [d'Aspremont et al., 2012] étudient le problème

de détection d'une composante principale parcimonieuse dans un modèle gaussien, i.e. testent la significativité des valeurs propres issues de l'ACP parcimonieuse. Plus précisément, [d'Aspremont et al., 2012] étudient le pouvoir discriminant de ces relaxations convexes entre deux modèles

$$\mathcal{N}(0, \mathbf{I}_n) \quad \text{and} \quad \mathcal{N}(0, \mathbf{I}_n + \theta vv^T)$$

où $v \in \mathbb{R}^n$ est un vecteur creux de norme euclidienne égale à un et de cardinalité k . [d'Aspremont et al., 2012] utilisent la relaxation développée dans [d'Aspremont et al., 2008] pour produire une statistique calculable en temps polynomial et montrent que dans un schéma où la dimension n , le nombre d'échantillons m et la cardinalité k du signal tendent vers l'infini proportionnellement, le seuil de détection θ reste borné. En particulier, [d'Aspremont et al., 2012] dérive une borne d'approximation pour la relaxation de l'ACP parcimonieuse [d'Aspremont et al., 2008] en s'inspirant des résultats de [Ben-Tal and Nemirovski, 2002]. Cette borne ne produit pas un ratio d'approximation constant et dépend de la valeur optimale du programme semidéfini : plus cette valeur est élevée, meilleure est la qualité d'approximation. Un comportement similaire avait été étudié par Zwick [1999] pour la relaxation semidéfinie du problème MaxCut, montrant que le ratio d'approximation de [Goemans and Williamson, 1995] pouvait être amélioré lorsque la valeur de la coupe optimale est suffisamment élevée. Les résultats de [d'Aspremont et al., 2012] montrent donc que dans certains régimes, la relaxation semidéfinie de [d'Aspremont et al., 2008] produit une bonne approximation de l'ACP parcimonieuse, et que le problème de détection se situe précisément dans l'un de ces régimes. Pour un problème de détection où

$$\rho = \frac{\Delta}{m} + \frac{\Delta}{\sqrt{k^* m (\Delta + 4/e)}} \quad \text{avec} \quad \Delta = 4 \log(9en/k^*) + 4 \log(1/\delta)$$

le seuil de détection minimax optimal est donné par

$$\theta_\phi = \left(2\sqrt{\frac{k^*(\Delta + 4/e)}{m}} + \frac{k^*(\Delta + 4/e)}{m} + 2\sqrt{\frac{\log(1/\delta)}{m}} \right) \left(1 - 2\sqrt{\frac{\log(1/\delta)}{m}} \right)^{-1}$$

en d'autres termes, en dessous de ce seuil, aucune statistique (combinatoire ou pas) ne pourra distinguer entre \mathcal{H}_0 et \mathcal{H}_1 . Supposons que $n = \mu m$ and $k^* = \kappa n$, où $\mu > 0$ et $\kappa \in (0, 1)$ sont fixés et n est suffisamment grand, [d'Aspremont et al., 2012] montrent que la relaxation semidéfinie permet la détection de composantes principales parcimonieuses au delà d'un seuil de détection θ_ψ satisfaisant

$$\theta_\psi = \beta(\mu, \kappa)^{-1} \theta_\phi$$

où $\beta(\mu, \kappa)$ est donné par

$$\beta(\mu, \kappa) = \frac{\vartheta(c)}{c} \quad \text{avec} \quad c = \frac{1 - \mu\Delta\kappa - \frac{\sqrt{\mu\kappa}}{\sqrt{(\Delta+4/e)}} - 2\sqrt{\frac{\log(1/\delta)}{m}}}{\mu\Delta + \frac{\mu\Delta}{\sqrt{\kappa(\Delta+4/e)}}},$$

où $\vartheta(1)$ est d'ordre un. Ceci signifie que le seuil de détection atteint par la statistique SDP (calculable en temps polynomial) reste fini lorsque la dimension n , le nombre d'échantillons m et la cardinalité du signal tendent vers l'infini proportionnellement

1.9 Sélection de covariance

Partant de données gaussiennes multivariées $x \sim \mathcal{N}(0, \Sigma)$ où $\Sigma \in \mathbf{S}^n$ est la covariance du modèle, une hypothèse structurelle classique permettant de réduire le nombre de paramètres à estimer consiste à

supposer que l’inverse de la matrice de covariance est creuse, i.e. qu’un grand nombre de ses coefficients sont égaux à zéro. Les zéros de la matrice inverse (appelée matrice de précision) correspondent à des paires de variables indépendantes, conditionnellement à toutes les autres. Pour estimer des matrices de covariance dont l’inverse est creuse, [Banerjee, El Ghaoui, and d’Aspremont, 2008] étudient le problème de maximum de vraisemblance pénalisé suivant

$$\max_{X \succ 0} \log \det X - \text{Tr}(CX) - \rho \sum_{ij} |X_{ij}|, \quad (7)$$

dans la variable $X \in \mathbf{S}_n$. Ce problème peut également être interprété comme un problème de maximisation de vraisemblance robuste avec bruit uniforme d’amplitude ρ sur les coefficients de la matrice de covariance empirique C . [d’Aspremont and El Ghaoui, 2006] dérivent deux algorithmes efficaces pour résoudre ce problème en grande dimension. Le premier est basé sur l’algorithme du premier ordre de [Nesterov, 1983] et a une complexité en $O(n^{4.5}/\epsilon)$ si l’on borne a priori le conditionnement de la solution. [d’Aspremont and El Ghaoui, 2006] dérivent également un algorithme d’optimisation par blocs qui exploite directement la structure du problème (7) pour le résoudre ligne/colonne par ligne/colonne. Chaque sous-problème est le dual d’un problème du type LASSO pour lequel des algorithmes très efficaces ont été développés. Une implémentation rapide en R de l’algorithme de [d’Aspremont and El Ghaoui, 2006], appelée GLASSO, a été produite par [Friedman et al., 2008].

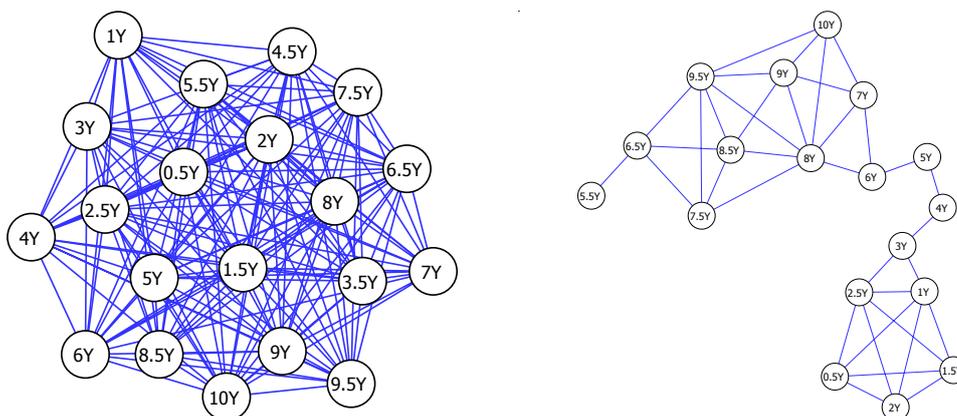


FIGURE 3: Réseau formé en utilisant le graphe de l’inverse de la solution de (7) pour une matrice de covariance de taux d’intérêt de maturités croissantes, pour $\rho = 0$ (gauche) et $\rho = 0.1$ (droite). Dans la solution parcimonieuse, la structure en chaîne de la dynamique de la “courbe” des taux apparaît clairement (graphique produit en utilisant cytoscape).

Pour illustrer ces techniques, la figure 3 représente la structure de dépendance pour la courbe des taux d’intérêt (échantillonnée sur un an), extraite de la matrice de covariance inverse. Chaque sommet du graphe représente un taux d’une certaine maturité et les arêtes correspondent à des coefficients non nuls dans la solution X de (7) (des variables conditionnellement dépendantes). Cette technique est aussi utilisée pour identifier des réseaux de gènes [Dobra et al., 2004] et la figure 4 montre son application a des données de vote pour le sénat américain.

1.10 Détection d’anomalies

[Cuturi, Vert, and d’Aspremont, 2009] utilisent la programmation semidéfinie pour formuler une relaxation convexe (exacte dans ce cas, grâce au théorème de Brickman) du problème d’extraction de

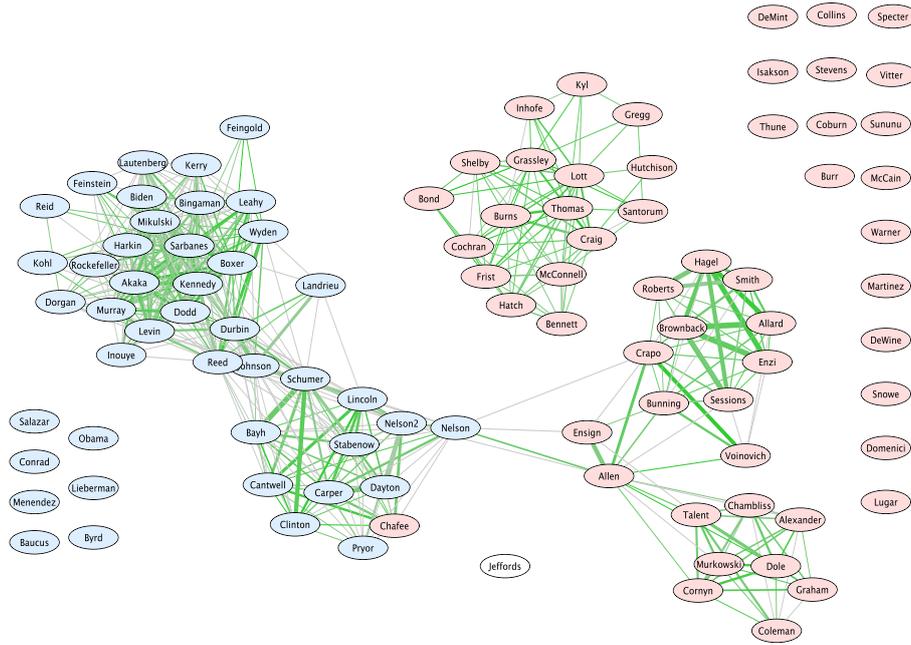


FIGURE 4: Représentation de la structure de corrélation des votes au 109^{ème} sénat américain. Dans ce graphe, l'épaisseur des liens croît avec la taille du coefficient de la matrice de précision. Les nuances de couleur représentent la confiance attachée à un lien, produite par bootstrapping, la fiabilité étant faible pour les liens gris et forte pour les liens colorés.

formes linéaires stationnaires à partir de données multivariées. Ces formes linéaires stables servent ensuite à la détection d'anomalies. A partir de la matrice de covariance $C \in \mathbf{S}_n$ et de l'opérateur d'auto-covariance $A \in \mathbb{R}^{n \times n}$, le problème s'écrit comme suit

$$\begin{aligned} & \text{maximize} && u^T C u, && \text{(high variance)} \\ & \text{subject to} && u^T A u = \mu, && \text{(low autocovariance)} \\ & && \|u\| = 1, && \text{(normalization)} \end{aligned}$$

qui est un programme quadratique non convexe dans le vecteur $u \in \mathbb{R}^n$. Le théorème de Brickman (voir [Brickman, 1961] ou [Barvinok, 2002, §II.13.4]) montre cependant que l'optimum de ce problème peut se calculer à partir de la solution de la relaxation semidéfinie

$$\begin{aligned} & \text{maximize} && \text{Tr}(CU) \\ & \text{subject to} && \text{Tr}(AU) = \mu \\ & && \text{Tr} U = 1, U \succeq 0, \end{aligned}$$

qui est un programme semidéfini en $U \in \mathbf{S}_n$ et peut être résolu efficacement. Ce problème est un exemple d'utilisation de la S-procédure issue du contrôle dans un contexte complètement différent.

1.11 Classification avec des noyaux indéfinis

Les machines à vecteurs support (SVM) sont un outil extrêmement flexible pour la classification de données en grande dimension. L'un des principaux attraits de ces méthodes est que toute l'information sur la géométrie des données est résumée dans une matrice de noyaux, qui traduit la similarité entre les différents points à classer. La seule contrainte qui régit la formation de cette matrice est qu'elle

soit représentable comme une matrice de Gram dans une espace bien choisi, ce qui lui impose d'être semidéfinie positive. Pour certains types de problèmes, les mesures de similarité utilisées en classification ne forment pas une matrice positive et les résultats de [Luss and d'Aspremont, 2008] ont cherché à étendre aux mesures de similarité les performances des SVM.

Si $K_0 \in \mathbf{S}_n$ est une matrice de noyaux et $y \in \mathbb{R}^n$ un vecteur de labels, le problème de calcul de la SVM s'écrit

$$\begin{aligned} & \text{maximize} && \alpha^T e - \text{Tr}(K_0(Y\alpha)(Y\alpha)^T)/2 \\ & \text{subject to} && 0 \leq \alpha \leq C, \alpha^T y = 0 \end{aligned}$$

dans la variable $\alpha \in \mathbb{R}^n$. [Luss and d'Aspremont, 2008] adaptent ce problème au cas où la matrice K_0 n'est pas positive en définissant un kernel "proxy" K , semidéfini positif, et en pénalisant la distance entre K_0 et K pour résoudre

$$\min_{\{K \succeq 0\}} \max_{\{\alpha^T y = 0, 0 \leq \alpha \leq C\}} \alpha^T e - \frac{1}{2} \text{Tr}(K(Y\alpha)(Y\alpha)^T) + \rho \|K - K_0\|_F^2$$

qui est un programme semidéfini dont les variables sont $K \in \mathbf{S}_n$ et $\alpha \in \mathbb{R}^n$, où le paramètre $\rho > 0$ contrôle la distance avec le noyaux initial. Le noyau optimal peut s'écrire comme

$$K^* = (K_0 + (Y\alpha)(Y\alpha)^T / (4\rho))_+$$

ce qui peut s'interpréter comme une perturbation du noyaux initial K_0 .

1.12 Calibration des modèles de taux

La calibration des modèles de taux revient à extraire une matrice de covariance des informations de prix sur un certain nombre de produits liquides. La stabilité et la fiabilité de ce processus ont un impact important sur la gestion des risques et la couverture des produits dérivés générés en utilisant le modèle. [d'Aspremont, 2003] dérive une formule d'approximation pour le prix des swaptions, les instruments de référence pour la corrélation sur les marchés de taux, et formule le problème de calibration comme un programme semidéfini.

1.13 Reconstruction de phase

Le problème de reconstruction de phase, i.e. le problème de reconstruire un signal complexe à partir d'informations sur le module d'observations linéaires, a de nombreuses applications en traitement du signal, notamment en cristallographie par rayons X [Harrison, 1993], en imagerie par diffraction [Bunk et al., 2007] ou en microscopie [Miao et al., 2008]. Dans toutes ces applications, les détecteurs ne mesurent que l'amplitude des signaux et pas leur phase. Des mesures linéaires sur un signal complexe $x \in \mathbb{C}^p$ sont obtenues via un opérateur A et le problème consiste à reconstruire x en n'utilisant que les valeurs des modules $|Ax|$. En fonction des propriétés de A , le problème de reconstruction de la phase de Ax peut avoir une solution unique caractérisée par les mesures $|Ax|$, et cette solution peut être stable sous certaines conditions.

Reconstruire la phase de Ax à partir des observations $|Ax|$ est un problème non convexe, combinatoire. Jusqu'à présent, ce problème était abordé en utilisant divers algorithmes gourmands [Gerchberg and Saxton, 1972, Fienup, 1982, Griffin and Lim, 1984] qui alternent projections sur l'image de A et sur l'ensemble des vecteur y tels que $|y| = |Ax|$. Plus récemment, [Chai, Moscoso, and Papanicolaou, 2011, Candes, Strohmer, and Voroninski, 2011a] ont formulé la reconstruction de phase comme un problème de complétion de matrice en utilisant une procédure de relaxation classique due à [Lovász and Schrijver, 1991], et ont étudié la performance de relaxations semidéfinies spécialisées à ce problème.

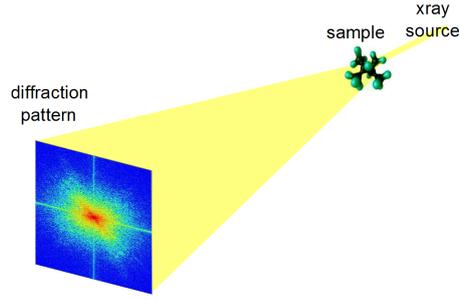


FIGURE 5: Reconstruction de phase et cristallographie par rayons X [Candes et al., 2011b].

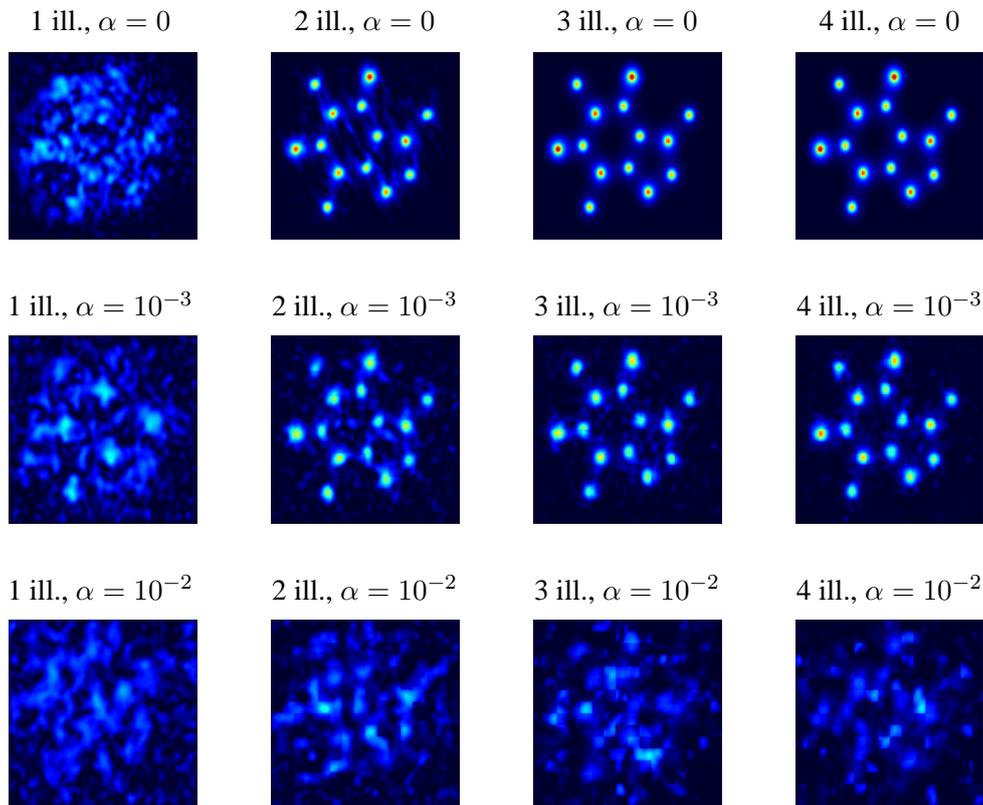


FIGURE 6: Solution du problème de reconstruction de phase en imagerie par diffraction de rayons X cohérents, obtenue à partir du SDP combiné avec un algorithme gourmand sur la molécule de caféine, pour plusieurs valeurs du nombre d'illuminations masquées et du niveau de bruit.

Les résultats récents de [Waldspurger, d'Aspremont, and Mallat, 2012] séparent le problème en phase du problème en signal et formulent le problème de reconstruction de phase comme un programme quadratique sur le tore complexe unité. La relaxation semidéfinie de ce programme a une structure très similaire à celle de la relaxation classique de MaxCut. Ce parallèle a un double avantage. Il permet d'abord d'appliquer la vaste librairie d'algorithmes pour MaxCut au problème de phase. Il permettra peut-être aussi de simplifier l'analyse de performance de la relaxation semidéfinie pour la reconstruction

de phase, en s’inspirant des nombreux résultats disponibles pour MaxCut. La figure 6 montre la solution d’un problème de reconstruction de phase en imagerie par diffraction de rayons X cohérents, obtenue à partir du SDP combiné avec un algorithme gourmand.

2 Dualité et Algorithmes

Cette section rappelle brièvement la construction du dual d’un programme semidéfini et évoque très brièvement aussi la résolution numériques de problèmes de taille raisonnable. Le lecteur est invité à consulter les ouvrages de [Boyd and Vandenberghe, 2004, Ben-Tal and Nemirovski, 2001] qui couvrent ces algorithmes plus en détail, ainsi que [Nesterov, 2003] sur les méthodes du premier ordre appliquées aux problèmes de grande dimension.

La dualité pour les programmes semidéfinis fonctionne exactement comme dans les cas des programmes linéaires, seul le cône change. Partant d’un SDP

$$\begin{aligned} & \text{minimize} && \mathbf{Tr}(CX) \\ & \text{subject to} && \mathbf{Tr}(A_i X) = b_i, \quad i = 1, \dots, m \\ & && X \succeq 0 \end{aligned}$$

dans la variable $X \in \mathbf{S}_n$. Le cône des matrices semidéfinies positives est symétrique, i.e.

$$Z \succeq 0 \iff \mathbf{Tr}(ZX) \geq 0, \text{ for all } X \succeq 0,$$

et les multiplicateurs de Lagrange sont donc des matrices semidéfinies positives. On forme le Lagrangien

$$L(X, y, Z) = \mathbf{Tr}(CX) + \sum_{i=1}^m y_i (b_i - \mathbf{Tr}(A_i X)) - \mathbf{Tr}(ZX)$$

avec multiplicateurs de Lagrange $y \in \mathbb{R}^m$ et $Z \in \mathbf{S}_n$ où $Z \succeq 0$. Après réarrangement des termes, on obtient

$$L(X, y, Z) = \mathbf{Tr}(X(C - \sum_{i=1}^m y_i A_i - Z)) + b^T y$$

En minimisant cette fonction affine de X , le dual s’écrit

$$\begin{aligned} & \text{maximize} && b^T y \\ & \text{subject to} && Z = C - \sum_{i=1}^m y_i A_i \\ & && Z \succeq 0, \end{aligned}$$

qui est également un programme semidéfini en y, Z .

La description des questions algorithmiques liées à la programmation semidéfinie sort du champs de cet article. La résolution de problèmes en grandes dimensions reste d’ailleurs un sujet de recherche largement ouvert. Les bibliothèques numériques destinées à résoudre des problèmes de petite taille en utilisant des méthodes de point intérieur sont par contre beaucoup plus mûres. Il existe même des libraires qui interprètent les problèmes d’optimisation conique “en langage naturel” et font l’interface avec les algorithmes classiques. En particulier, la relaxation SDP du problème MAXCUT

$$\begin{aligned} & \text{max.} && \mathbf{Tr}(XC) \\ & \text{s.t.} && \mathbf{diag}(X) = \mathbf{1} \\ & && X \succeq 0, \end{aligned}$$

se résout simplement en utilisant CVX [Grant et al., 2001] et SDPT3 [Toh et al., 1999], avec le code suivant

```

cvx_begin
variable X(n,n) symmetric
maximize trace(C*X)
subject to
    diag(X)==1
    X==semidefinite(n)
cvx_end

```

Ce programme résout très efficacement des problèmes comportant une centaine de variables. Au delà de quelques milliers de variables, la question reste ouverte.

Références

- F. Alizadeh. Interior point methods in semidefinite programming with applications to combinatorial optimization. *SIAM Journal on Optimization*, 5 :13–51, 1995.
- N. Alon and A. Naor. Approximating the cut-norm via Grothendieck’s inequality. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 72–80. ACM, 2004.
- A.A. Amini and M. Wainwright. High-dimensional analysis of semidefinite relaxations for sparse principal components. *The Annals of Statistics*, 37(5B) :2877–2921, 2009.
- O. Banerjee, L. El Ghaoui, and A. d’Aspremont. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *The Journal of Machine Learning Research*, 9 :485–516, 2008.
- A. Barvinok. *A course in convexity*. American Mathematical Society, 2002.
- A. Ben-Tal and A. Nemirovski. *Lectures on modern convex optimization : analysis, algorithms, and engineering applications*. MPS-SIAM series on optimization. Society for Industrial and Applied Mathematics : Mathematical Programming Society, Philadelphia, PA, 2001.
- A. Ben-Tal and A. Nemirovski. On tractable approximations of uncertain linear matrix inequalities affected by interval uncertainty. *SIAM Journal on Optimization*, 12(3) :811–833, 2002.
- A. Ben-Tal, A. Nemirovski, and C. Roos. Robust solutions of uncertain quadratic and conic-quadratic problems. *SIAM Journal on Optimization*, 13(2) :535–560, 2003. ISSN 1052-6234.
- A. Ben-Tal, L. El Ghaoui, and A.S. Nemirovski. *Robust optimization*. Princeton University Press, 2009.
- Q. Berthet and P. Rigollet. Optimal detection of sparse principal components in high dimension. *Arxiv preprint arXiv :1202.5070*, 2012.
- S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- S. Boyd, P. Diaconis, and L. Xiao. Fastest mixing Markov chain on a graph. *Siam Review*, 46(4) :667–690, 2004.
- L. Brickman. On the field of values of a matrix. *Proceedings of the American Mathematical Society*, pages 61–66, 1961.
- O. Bunk, A. Diaz, F. Pfeiffer, C. David, B. Schmitt, D.K. Satapathy, and JF Veen. Diffractive imaging for periodic samples : retrieving one-dimensional concentration profiles across microfluidic channels. *Acta Crystallographica Section A : Foundations of Crystallography*, 63(4) :306–314, 2007.
- E. J. Candes, T. Strohmer, and V. Voroninski. Phaselift : exact and stable signal recovery from magnitude measurements via convex programming. *To appear in Communications in Pure and Applied Mathematics*, 2011a.
- E.J. Candes, Y. Eldar, T. Strohmer, and V. Voroninski. Phase retrieval via matrix completion. *Arxiv preprint arXiv :1109.0573*, 2011b.
- A. Chai, M. Moscoso, and G. Papanicolaou. Array imaging using intensity-only measurements. *Inverse Problems*, 27 :015005, 2011.

- M. Cuturi, J.-P. Vert, and A. d'Aspremont. White functionals for anomaly detection in dynamical systems. *ArXiv : 0908.0143*, 2009.
- G. B. Dantzig. *Linear programming and extensions*. Princeton University press, 1963.
- A. d'Aspremont. Interest rate model calibration using semidefinite programming. *Applied Mathematical Finance*, 10(3) :183–213, 2003.
- A. d'Aspremont and L. El Ghaoui. A semidefinite relaxation for air traffic flow scheduling. *ArXiv cs.CE/0609145*, 2006.
- A. d'Aspremont, L. El Ghaoui, M.I. Jordan, and G. R. G. Lanckriet. A direct formulation for sparse PCA using semidefinite programming. *SIAM Review*, 49(3) :434–448, 2007.
- A. d'Aspremont, F. Bach, and L. El Ghaoui. Optimal solutions for sparse principal component analysis. *Journal of Machine Learning Research*, 9 :1269–1294, 2008.
- A. d'Aspremont, F. Bach, and L. El Ghaoui. Approximation bounds for sparse principal component analysis. *ArXiv : 1205.0121*, 2012.
- A. Dobra, C. Hans, B. Jones, J.R. J. R. Nevins, G. Yao, and M. West. Sparse graphical models for exploring gene expression data. *Journal of Multivariate Analysis*, 90(1) :196–212, 2004.
- J.R. Fienup. Phase retrieval algorithms : a comparison. *Applied Optics*, 21(15) :2758–2769, 1982.
- J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3) :432, 2008.
- A. Frieze and R. Kannan. Quick approximation to matrices and applications. *Combinatorica*, 19(2) :175–220, 1999.
- R. Gerchberg and W. Saxton. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35 :237–246, 1972.
- M.X. Goemans and D.P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42 :1115–1145, 1995.
- M. Grant, S. Boyd, and Y. Ye. *CVX : Matlab software for disciplined convex programming*. 2001.
- D. Griffin and J. Lim. Signal estimation from modified short-time fourier transform. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 32(2) :236–243, 1984.
- R.W. Harrison. Phase problem in crystallography. *JOSA A*, 10(5) :1046–1055, 1993.
- I. T. Jolliffe. *Principal component analysis*. Springer-Verlag, 1986.
- N. K. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4 :373–395, 1984.
- L. G. Khachiyan. A polynomial algorithm in linear programming (in Russian). *Doklady Akedamii Nauk SSSR*, 244 :1093–1096, 1979.
- J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIAM Journal on Optimization*, 11(3) :796–817, 2001.
- L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal on Optimization*, 1(2) :166–190, 1991.
- R. Luss and A. d'Aspremont. Support vector machine classification with indefinite kernels. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 953–960. MIT Press, Cambridge, MA, 2008.
- J. Miao, T. Ishikawa, Q. Shen, and T. Earnest. Extending x-ray crystallography to allow the imaging of noncrystalline materials, cells, and single protein complexes. *Annu. Rev. Phys. Chem.*, 59 :387–410, 2008.
- A.S. Nemirovski. *Computation of matrix norms with applications to Robust Optimization*. PhD thesis, Technion, 2005.
- A. Nemirovskii and D. Yudin. Problem complexity and method efficiency in optimization. *Nauka (published in*

English by John Wiley, Chichester, 1983), 1979.

- Y. Nesterov. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, 27(2) :372–376, 1983.
- Y. Nesterov. Semidefinite relaxation and nonconvex quadratic optimization. *Optimization methods and software*, 9(1) :141–160, 1998.
- Y. Nesterov. Squared functional systems and optimization problems. Technical Report 1472, CORE reprints, 2000.
- Y. Nesterov. *Introductory Lectures on Convex Optimization*. Springer, 2003.
- Y. Nesterov and A. Nemirovskii. *Interior-point polynomial algorithms in convex programming*. Society for Industrial and Applied Mathematics, Philadelphia, 1994.
- P. Parrilo. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, 2000.
- M. Putinar. Positive polynomials on compact semi-algebraic sets. *Indiana University Mathematics Journal*, 42(3) :969–984, 1993.
- N.Z. Shor. Quadratic optimization problems. *Soviet Journal of Computer and Systems Sciences*, 25 :1–11, 1987.
- J. Sun, S. Boyd, L. Xiao, and P. Diaconis. The fastest mixing Markov process on a graph and a connection to a maximum variance unfolding problem. *SIAM Review*, 48(4) :681–699, 2006.
- K. C. Toh, M. J. Todd, and R. H. Tutuncu. SDPT3 – a MATLAB software package for semidefinite programming. *Optimization Methods and Software*, 11 :545–581, 1999.
- I. Waldspurger, A. d’Aspremont, and S. Mallat. Phase recovery, maxcut and complex semidefinite programming. *ArXiv : 1206.0102*, 2012.
- K.Q. Weinberger and L.K. Saul. Unsupervised Learning of Image Manifolds by Semidefinite Programming. *International Journal of Computer Vision*, 70(1) :77–90, 2006.
- U. Zwick. Outward rotations : a tool for rounding solutions of semidefinite programming relaxations, with applications to max cut and other problems. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 679–687. ACM, 1999.