TECHNIQUES D'OPTIMISATION: IFT 3512 22 mars 2004

CHAPITRE 1. GÉNÉRALITÉS. OPTIMISATION SANS CONTRAINTES. PROGRAMMATION LINÉAIRE

Notions à revoir: ensemble (ouvert, fermé, connexe, borné, compact, frontière, intérieur, point d'accumulation (point limite), intersection, union, somme, produit cartésien), **espace euclidien** (distance, norme, produit scalaire), **suite** (limite, limite supérieure, limite inférieure), **algèbre linéaire** (matrice, valeur propre, matrice définie positive, matrice définie négative), **calcul** (continuité, différentiabilité, fonction composée (règle de dérivation), dérivée directionnelle, jacobien, hessien).

1. **problème générique:** $\min_{x \in X} f(x)$ où $X \subset R^n$ et $f(x) \in R$.

   minimum local x^***:** $\exists \epsilon > 0$ tel que $f(x) - f(x^*) \geq 0 \quad \forall x \neq x^*$ tel que $\|x - x^*\| < \epsilon$.

   minimum global x^***:** $f(x) - f(x^*) \geq 0 \quad \forall x \neq x^*$.

   minimum local strict: remplacer "$\geq$" ci-dessus par "$>$".

   Une fonction est **unimodale** si tout minimum local est global.

2. Un ensemble X est **convexe** si $x, y \in X \;\Rightarrow\; [x, y] \subset X$, c-à-d: $\theta x + (1-\theta)y \in X$ pour tout $x, y \in X$ et $\theta \in [0, 1]$. L'intersection, la somme et le produit cartésien d'ensembles convexes sont convexes. L'union d'ensembles convexes n'est en général *pas* convexe. L'**enveloppe convexe** d'un ensemble X est le plus petit ensemble convexe contenant X, c'est-à-dire l'intersection de tous les ensembles convexes contenant X. Un **polyèdre** $P = \{x : Ax \leq b\}$ est l'intersection d'un nombre fini de **demi-espaces**. Un **polytope** est l'enveloppe convexe d'un nombre fini de points. Tout polytope est un polyèdre. Tout point de l'enveloppe convexe d'un nombre fini de points de R^n peut s'exprimer comme une combinaison convexe d'au plus $n + 1$ des points originaux.

3. Une fonction f définie sur un ensemble convexe X est convexe sur X si (et seulement si)

   $$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y) \quad \forall x, y \in X \quad \forall \theta \in (0, 1).$$

   Toute fonction convexe définie sur un ensemble ouvert est continue et différentiable (presque partout). Toute fonction convexe est unimodale. Une fonction est **strictement convexe** si l'inégalité est stricte pour tout $x \neq y$. Une fonction est **fortement convexe** s'il existe une constante positive β telle que

   $$f(\theta x + (1-\theta)y) \leq \theta f(x) + (1-\theta) f(y) - \beta\theta(1-\theta)\|x - y\|^2 \quad \forall x, y \in X \quad \forall \theta \in (0, 1).$$

   Si f est différentiable sur l'ensemble convexe X, alors les deux propriétés suivantes sont équivalentes à la convexité de f sur X:

   inégalité du gradient: $\quad f(y) \geq f(x) + \nabla f(x)(y - x) \quad \forall x, y \in X$

   monotonie: $\quad (\nabla f(x) - \nabla f(y))(x - y) \geq 0 \quad \forall x, y \in X.$

   Si l'inégalité est stricte pour $x \neq y$ (**monotonie stricte**), la fonction f est strictement convexe. Dans la cas de la forte convexité, on rajoute le terme $\beta\|x - y\|^2$ au membre de droite des deux inégalités. On parle alors de **forte monotonie**.

   Une fonction deux fois différentiable est convexe si et seulement si $(x - y)^t \nabla^2 f(x)(x - y) \geq 0 \quad \forall x, y \in X$. Si $X = R^n$, cette condition revient à dire que la matrice hessienne $H(x) = \nabla^2 f(x)$ est semidéfinie positive sur X. Si la matrice hessienne est uniformément définie positive pour tout x dans X (ses valeurs propres sont bornées inférieurement par un nombre positif), alors la fonction f est fortement convexe. Toute combinaison linéaire non négative de fonctions convexes est convexe. e maximum d'un ensemble de fonctions convexes est une fonctionconvexe. Une fonction f est **concave** si $-f$ est convexe.

4. **développement de Taylor:**

   premier ordre: $f(y) = f(x) + \nabla f(x)(y - x) + o(\|y - x\|) = f(x) + \nabla f(\xi)(y - x)$ où $\xi \in [x, y]$.

   deuxième ordre:

   $$\begin{aligned} f(y) &= f(x) + \nabla f(x)(y - x) + \frac{1}{2}(y - x)^t H(x)(y - x) + o(\|y - x\|^2) \\ &= f(x) + \nabla f(x)(y - x) + \frac{1}{2}(y - x)^t H(\xi)(y - x) \qquad \text{où } \xi \in [x, y]. \end{aligned}$$

5. **conditions nécessaires d'optimalité du premier ordre:**

   $g'(x^*; d) = \nabla f(x^*)d \geq 0$ pour toute direction réalisable d. Si $X = R^n$, ces conditions se résument à: $\nabla f(x^*) = 0$.

   Un point satisfaisant les conditions d'optimalité du premier ordre est appelé **point stationnaire**.

   Tout point stationnaire d'une fonction convexe est un minimum *global*.

6. **conditions nécessaires d'optimalité du second ordre:** On rajoute aux conditions du premier ordre la condition:
$$d^t H(x^*) d \geq 0 \quad \text{si} \quad \nabla f(x^*) d = 0.$$
   Si $X = R^n$ cette dernière condition se résume à: $H(x^*)$ semidéfinie positive.

7. Un **algorithme** est une multifonction qui associe à un point x^k de R^n un point x^{k+1} du sous-ensemble $A(x^k)$ de R^n. Dénotons par X^* l'ensemble des solutions. Un algorithme est un **algorithme de descente** pour la fonction continue Z définie sur X si:
$$\begin{array}{lllll} \text{(i)} & x \in X & \text{et} & y \in A(x) \Rightarrow & y \in X \\ \text{(ii)} & x \notin X^* & \text{et} & y \in A(x) \Rightarrow & Z(y) < Z(x) \\ \text{(iii)} & x \in X^* & \text{et} & y \in A(x) \Rightarrow & Z(y) \leq Z(x). \end{array}$$
   Un algorithme est **fermé** si
$$\left.\begin{array}{l} x^k \in X \\ x^k \rightarrow x \\ y^k \in A(x^k) \\ y^k \rightarrow y \end{array}\right\} \Rightarrow \; y \in A(x).$$
   Une multifonction A est fermée si et seulement si son **graphe** $G = \{(x,u) | x \in X, u \in A(x)\}$ est un ensemble fermé.

   théorème de convergence globale: Soit $\{x^k\}$ une suite engendrée par un algorithme de descente. Si X est compact et $A(x)$ est fermé pour tout $x \notin X^*$, alors la limite de toute sous-suite convergente appartient à l'ensemble des solutions X^*.

   La choix de la fonction de descente Z est souvent $f(x^k)$ ou $\|x^k - x^*\|^2$. Sauf indication contraire, tous les algorithmes mentionnés dans ces notes sont fermés.

8. **ordre de convergence** τ d'une suite $\{r_k\}$ vers le nombre r^*:
$$\tau \stackrel{\text{def}}{=} \limsup_{p \geq 0} \left\{ \beta = \limsup_{k \to \infty} \frac{|r_{k+1} - r^*|}{|r_k - r^*|^p} < \infty \right\}.$$
   Si $p = 1$ et $0 < \beta < 1$, on dit que la suite converge **linéairement** (ou **géométriquement**) vers r^* avec un **taux de convergence** égal à β. Si la suite converge et $\beta = 1$ on dit que la convergence est **sous-linéaire**.

   Si $p = 2$ et $0 < \beta < \infty$, on dit que la convergence est **quadratique**.

   Si $p = 1$ et $\beta = 0$ on dit que la convergence est **superlinéaire**.

9. **Optimisation d'une fonction unidimensionnelle unimodale définie sur un intervalle:**

   Étant donnés un intervalle $[x_0, x_{N+1}] = [a, b]$ et deux points intérieurs x_1 et x_2 ($x_1 < x_2$), la connaissance des valeurs de la fonction f en ces deux points permet de réduire l'intervalle d'incertitude: si $f(x_1) > f(x_2)$ on conserve l'intervalle $[x_1, x_{N+1}]$, sinon on conserve l'intervalle $[x_0, x_2]$.

   Supposons que l'intervalle $[x_1, x_{N+1}]$ soit retenu. Une stratégie "naturelle" consiste à choisir le prochain point d'évaluation symétriquement à x_2 par rapport à x_1 et x_{N+1}. Il en va de même si l'intervalle $[x_0, x_2]$ est retenu. Le choix des points d'évaluation est alors déterminé par l'emplacement du point d'évaluation initial x_1 et les valeurs prises par f aux points d'évaluation successifs. Étant donné un nombre de points d'évaluation N connu a priori, le choix permettant de réduire au minimum la largeur de l'intervalle d'incertitude est: $x_1 = a + (F_{N-1}/F_N)(b-a)$, où F_N est le Nième terme de la **suite de Fibonacci** définie récursivement par: $F_0 = F_1 = 1$, $F_N = F_{N-1} + F_{N-2}$. A l'itération k, la largeur de l'intervalle d'incertitude est $(F_{N-k+1}/F_N)(b-a)$. A la dernière itération, plutôt que de faire coïncider x_{N-1} et x_N, (ce qui ne donnerait aucune information supplémentaire) on situe x_N arbitrairement proche de x_{N-1}, de façon à réduire de moitié la largeur de l'intervalle.

Puisque $F_N = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^N - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^N$, on a: $\lim_{N\to\infty} \frac{F_{N+1}}{F_N} = (1+\sqrt{5})/2 = \phi$ (le **nombre d'or** des Grecs). Le taux de convergence asymptotique est égal à $1/\phi$.

On peut également choisir $x_1 = a + (b-a)/\phi$. Ce choix est indépendant de N et reproduit les mêmes rapports entre les intervalles, modulo un changement d'échelle. Après k itérations, l'intervalle d'incertitude est réduit par un facteur ϕ^k. Cette méthode de la **section dorée** est asymptotiquement aussi efficace que la méthode de Fibonacci. À chaque itération, les proportions sont conservées, c-à-d, si $a = 0$ et $b = 1$: $x_1/1 = x_2/x_1 = (1-x_1)/(1-x_2)$.

10. **recherche binaire:** si la fonction f est différentiable et que $f'(a) < 0$ et $f'(b) > 0$, on peut déterminer un zéro de f', c'est-à-dire un point stationnaire de f. Cette technique réduit de moitié l'intervalle d'incertitude à chaque itération en évaluant f' au milieu de l'intervalle: $x_{k+1} = (x_k + x_{k-1})/2$. Si $f'(x) < 0$, au moins un zéro doit se trouver à droite de x; si $f'(x) > 0$, au moins un zéro doit se trouver à gauche de x.

11. La **méthode de Newton unidimensionnelle** minimise l'approximation quadratique (Taylor au second ordre) de la fonction f au point courant. La méthode, comme la recherche binaire, peut s'appliquer à la recherche d'une racine d'une fonction g. Elle consiste alors à trouver le zéro de l'approximation *linéaire* de g au point courant, c'est-à-dire à résoudre l'équation:

$$g(x_k) + g'(x_k)(x - x_k) = 0,$$

dont la solution est: $x_{k+1} = x_k - g(x_k)/g'(x_k)$. Si $g(x_*) = 0$ et $g'(x_*) \neq 0$, la convergence de la méthode vers x_* est localement quadratique.

12. La méthode de la **sécante** cherche une racine de l'équation $g(x) = 0$ en approximant la fonction g par une droite joignant les deux points les plus récents. Cette droite coupe l'axe des abscisses au point:

$$x_{k+1} = x_k - g(x_k)\frac{x_k - x_{k-1}}{g(x_k) - g(x_{k-1})}.$$

Si $g(x_*) = 0$ et $g'(x_*) \neq 0$ la convergence locale de la méthode est d'ordre $1 + \phi = 1/\phi$.

La méthode de la **fausse position** (regula falsi) cherche une racine de l'équation $g(x) = 0$ sur l'intervalle $[a, b]$, où $g(a) < 0$ et $g(b) > 0$, en approximant la fonction g par une droite joignant les deux points les plus récents *ayant des valeurs opposées*. La convergence peut être en pratique nettement moins bonne que celle de la recherche dichotomique.

13. **ajustement quadratique:** à l'itération k, on minimise le polynôme d'interpolation passant par les points $(x_{k-1}, f(x_{k-1}))$, $(x_{k-2}, f(x_{k-2}))$ et $(x_{k-3}, f(x_{k-3}))$. L'ordre de convergence de la méthode est d'environ 1.3.

14. **méthode de la plus grande pente**, ou **méthode du gradient:** on pose $g(x) = \nabla f(x)^t$ et on choisit: $x^{k+1} = x^k - \alpha_k g(x^k)$ où $\alpha_k \in \arg\min_{\alpha\geq 0} f(x^k - \alpha g(x^k))$. Si f est la fonction quadratique $f(x) = (1/2)x^tQx$, où Q est une matrice symétrique définie positive dont la plus petite valeur propre est a et la plus grande valeur propre est A, on a l'**inégalité de Kantorovitch**:

$$\frac{(x^tx)^2}{(x^tQx)(x^tQ^{-1}x)} \geq \frac{4aA}{(a+A)^2}.$$

On en déduit:

$$f(x^{k+1}) \leq \left(\frac{A-a}{A+a}\right)^2 f(x^k).$$

Dans le cas général, la convergence est linéaire de taux $\left(\frac{A-a}{A+a}\right)^2$.

15. **méthode de Newton multidimensionnelle:** on généralise la méthode de Newton unidimensionnelle, qui devient maintenant:

$$x^{k+1} = x^k - [J(x^k)]^{-1}g(x^k),$$

où $J(x^k)$ dénote la matrice jacobienne de g évaluée au point x^k. Si la fonction g est le gradient d'une fonction convexe f, alors la direction de Newton est une direction de descente pour f, et on peut rendre la méthode globalement convergente en effectuant, à chaque itération, une recherche dans cette direction.

16. **Gauss-Seidel:** on minimise la fonction f pour chaque variable séquentiellement, de la première à la dernière, puis on recommence à la première, etc., ce qui donne, à chaque itération majeure:

$$x_l^{k+1} \in \arg\min_x f(x_1^{k+1}, x_2^{k+1}, \ldots, x_{l-1}^{k+1}, x, x_{l+1}^k, \ldots, x_n^k), \quad l = 1, \ldots, n.$$

On peut également appliquer cette technique à la résolution d'un système d'équations $F(x) = 0$, ce qui donne:

$$F_l(x_1^{k+1}, x_2^{k+1}, \ldots, x_{l-1}^{k+1}, x_l^{k+1}, x_{l+1}^k, \ldots, x_n^k) = 0, \quad l = 1, \ldots, n.$$

La méthode de **Jacobi** est similaire, mais agit en parallèle sur toutes les variables:

$$x_l^{k+1} \in \arg\min_x f(x_1^k, x_2^k, \ldots, x_{l-1}^k, x, x_{l+1}^k, \ldots, x_n^k), \quad l = 1, \ldots, n.$$

et

$$F_l(x_1^k, x_2^k, \ldots, x_{l-1}^k, x_l^k, x_{l+1}^k, \ldots, x_n^k) = 0, \quad l = 1, \ldots, n.$$

Ces méthodes sont des cas particuliers d'approximation de la fonction originelle F par une fonction G plus simple à manipuler: $F(x) \approx G(x, x^k)$ avec $G(x^k, x^k) = F(x^k)$. Si la fonction $F = Ax + b$ est affine, la méthode de Gauss-Seidel résoud le système $Lx + (A - L)x^k = 0$, où L est la matrice triangulaire contenant les éléments de A situés sur ou sous la diagonale principale. La méthode de Jacobi résoud le système diagonal $Dx + (A - D)x^k = 0$, où D est la diagonale de la matrice A. Si la matrice A est définie positive, les deux méthodes convergent vers la solution du système.

17. **critères d'arrêt d'une minimisation unidimensionnelle:** soit d une direction de descente admissible pour la fonction f au point x^k. On définit la fonction unidimensionnelle

$$\varphi(\alpha) = f(x + \alpha d).$$

On arrête la minimisation de φ lorsqu'un taux de décroissance suffisant a été obtenu, c'est-à-dire

$$\varphi(\alpha) \leq \varphi(0) + \epsilon\varphi'(0)\alpha$$

pour un certain paramètre $0 < \epsilon < 1$ et que:

règle d'Armijo: (α assez grand) $\boxed{\varphi(\eta\alpha) > \varphi(0) + \epsilon\varphi'(0)\eta\alpha}$ où $\eta > 1$.

règle de Goldstein: $0 < \epsilon < 1/2$ et α assez grand: $\boxed{\varphi(\alpha) > \varphi(0) + (1 - \epsilon)\varphi'(0)\alpha}$.

règle de Wolfe: $0 < \epsilon < 1/2$ et α assez grand: $\boxed{\varphi'(\alpha) \geq (1 - \epsilon)\varphi'(0)}$. Ce dernier critère est invariant par rapport aux changements d'échelle.

En pratique on part d'un α assez grand, qu'on diminue par un facteur constant jusqu'à ce que le critère soit satisfait. Si les critères d'Armijo, Goldstein ou Wolfe sont couplés à des méthodes engendrant des directions de recherche fermées, les algorithmes résultant sont globalement convergents.

18. **méthodes de directions conjuguées.** Soit le problème quadratique convexe

$$\min f(x) = \frac{1}{2}x^t Q x + b^t x$$

où la matrice $Q : n \times n$ est définie positive. Pour tout x on pose

$$g(x) = \nabla f(x)^t = Qx + b \qquad g_k = g(x^k).$$

On considère un ensemble de **directions Q-conjuguées**[1]:$\{p_0, \ldots, p_k\}$, c'est-à-dire:

$$p_i^t Q p_j \neq 0 \iff i = j$$

ainsi que la variété affine $H_k = x^k + \text{lin}\,\{p_0, \ldots, p_k\}$. Soit $P_k = (p_0, \ldots, p_k)$. On a:

$$\min_{x \in H_k} f(x) \equiv \min_{w \in R^{k+1}} f(x^k + P_k w) = f(x^k + \sum_{i=0}^k w_i p_i)$$

[1]Par exemple, les directions correspondant aux vecteurs propres de la matrice Q (axes principaux) sont Q-conjuguées.

avec

$$\begin{aligned} f(x^k+P_kw) &= \frac{1}{2}(x^k+P_kw)^tQ(x^k+P_kw)+b^t(x^k+P_kw) \\ &= \frac{1}{2}w^tP_k^tQP_kw+g_k^tP_kw+f(x^k) \\ &= \sum_{i=0}^{k}\Big[\frac{1}{2}(p_i^tQp_i)w_i^2+(g_k^tp_i)w_i\Big]+f(x^k). \end{aligned}$$

qui est une fonction séparable, simple à minimiser. Son minimum est atteint en $w_k = -(g_k^tp_k)/(p_k^tQp_k)$. Comme x^k est le minimum de la fonction f sur la variété H_k et que $H_{n-1} = R^n$, la solution optimale est obtenu après exactement n itérations. En utilisant la notation α_k plutôt que w_k pour le pas dans la direction p_k, on obtient:

$$x^{k+1} = x^k+\alpha_kp_k \quad \text{avec} \quad \alpha_k = -\frac{g_k^tp_k}{p_k^tQp_k}.$$

Pour l'algorithme du **gradient conjugué** on fait le choix: $p_k = -g_k+\sum_{j=0}^{k-1}\beta_{kj}p_j$. Puisque $g_i \in \text{lin}\,\{p_0,\ldots,p_k\}$ pour tout $i < k$, on en déduit que $g_k^tg_i = 0$ pour tout $i < k$. Quelques calculs élémentaires (processus d'orthogonalisation de Gram-Schmidt[2]) permettent d'obtenir la formule:

$$p_k = -g_k+\sum_{j=0}^{k-1}\frac{g_k^tQp_j}{p_j^tQp_j}p_j.$$

Notons que $g_{j+1}-g_j = Q(x^{j+1}-x^j) = \alpha_jQp_j \quad \Rightarrow \quad g_k^tQp_j = g_k^t(g_{j+1}-g_j)/\alpha_j \neq 0 \Leftrightarrow j = k-1$, ce qui mène à l'expression simplifiée:

$$p_k = -g_k+\frac{g_k^t(g_k-g_{k-1})}{\alpha_{k-1}p_{k-1}^tQp_{k-1}}p_{k-1}.$$

Enfin, puisque $\alpha_{k-1} = -g_{k-1}^tp_{k-1}/p_{k-1}^tQp_{k-1} = \|g_{k-1}\|^2/p_{k-1}^tQp_{k-1} \quad \Rightarrow \quad \alpha_{k-1}p_{k-1}^tQp_{k-1} = \|g_{k-1}\|^2$, on obtient

$$p_k = -g_k+\beta_{k-1}p_{k-1}$$

où le coefficient β_{k-1} peut s'exprimer sous la forme $\boxed{\beta_{k-1} = (g_k-g_{k-1})^tg_k/\|g_{k-1}\|^2}$ ou $\boxed{\beta_{k-1} = \|g_k\|^2/\|g_{k-1}\|^2}$.

ALGORITHME (gradient conjugué)	x^0 arbitraire
$\beta_{-1} = 0 \qquad p_{-1} = 0$	
$p_k = -g_k+\beta_{k-1}p_{k-1}$	$\alpha_k = \|g_k\|^2/p_k^tQp_k$
$x^{k+1} = x^k+\alpha_kp_k$	$\beta_k = \|g_{k+1}\|^2/\|g_k\|^2 = (g_{k+1}-g_k)^tg_k/\|g_k\|^2$

Dans le cas non linéaire, on identifie g_k à $\nabla f(x^k)^t$ et on utilise soit l'expression $\beta_{k-1} = \|g_k\|^2/\|g_{k-1}\|^2$ (**Fletcher-Reeves**), soit l'expression $\beta_{k-1} = (g_k-g_{k-1})^t/\|g_{k-1}\|^2$ (**Polak-Ribière**). La valeur de α_k est obtenue à partir de la recherche linéaire

$$\min_{\alpha\geq 0} f(x^k+\alpha p_k).$$

Puisque le minimum n'est pas toujours atteint en n itérations, l'algorithme procède par cycles de n sous-itérations. La méthode du gradient conjugué est à peine plus coûteuse à implanter que l'algorithme du gradient, et beaucoup plus efficace.

[2]Si $\{b_k\}$ est un ensemble de vecteurs linéairement indépendants, on obtient un ensemble de vecteurs orthogonaux définissant le même sous-espace à l'aide de la formule $b'_k = \text{norm}\{b_k-\sum_{i=0}^{k-1}\langle b_k,b'_i\rangle b'_i\}$, où $\text{norm}(x) = x/\|x\|$.

19. **Méthodes quasi-Newton**

Comme l'algorithme du gradient conjugué, cette classe de méthodes cherche à tirer le meilleur profit possible de l'information de premier ordre (gradient) fournie par les itérés. À chaque itération, on résoud un système linéaire dont la matrice B_k est une approximation de la matrice hessienne $H(x^k)$. Le schéma de l'algorithme est simple:

ALGORITHME (quasi-Newton) $\quad x^0$ arbitraire et B_0 définie positive
itération k: résoudre $B_k d = -\nabla f(x^k) \to d^k$
$\min_\alpha f(x^k + \alpha d^k) \to x^{k+1}$
mise-à-jour de B_k

On souhaite que la matrice B_k possède les propriétés suivantes:

- facile à mettre à jour,
- symétrique,
- "converge" vers la matrice hessienne;
- généralise la méthode de la sécante, c'est-à-dire possède la "propriété de la sécante" (exacte dans le cas quadratique)

$$\boxed{B_k(x^k - x^{k-1}) = g_k - g_{k-1}} \; .$$

- résolve un problème quadratique de dimension n en n itérations;
- soit définie positive.

En posant $p_k = x^{k+1} - x^k$ et $u_k = g_{k+1} - g_k$, la propriété de la sécante se résume à $B_{k+1} p_k = u_k$. Une **correction de rang un** qui conserve la symétrie est obtenue en posant $\boxed{B_{k+1} = B_k + \beta z z^t}$. On déduit de la propriété de la sécante la relation $B_k p_k + \beta z z^t p_k = u_k$. On a:

$$(u_k - B_k p_k)(u_k - B_k p_k)^t = \beta^2 p_k^t z z^t z z^t p_k = \beta^2 (p_k^t z)^2 z z^t \quad \Rightarrow \quad z z^t = \frac{(u_k - B_k p_k)(u_k - B_k p_k)^t}{\beta^2 (p_k^t z)^2}.$$

Or, en prenant le produit scalaire avec p_k dans l'égalité de la sécante, on obtient

$$\begin{aligned} p_k^t B_k p_k + \beta p_k^t z z^t p_k = p_k^t u_k \quad & \Rightarrow \quad \beta (p_k^t z)^2 = p_k^t (u_k - B_k p_k) \\ & \Rightarrow \quad B_{k+1} = B_k + \frac{(u_k - B_k p_k)(u_k - B_k p_k)^t}{p_k^t (u_k - B_k p_k)} \; . \end{aligned}$$

Il est également possible d'obtenir des formules qui mettent à jour l'*inverse* de la matrice hessienne. Malheureusement, la matrice B_k, dans les deux cas, n'est pas nécessairement définie positive. Pour cela, il faut utiliser (au moins) une mise-à-jour de rang deux. La formule suivante, qui définit la **famille de Broyden**, possède toutes les propriétés requises, lorsque couplée avec une recherche linéaire adéquate:

$$\begin{aligned} B_{k+1} &= B_k - \frac{(B_k p_k)(B_k p_k)^t}{p_k^t B_k p_k} + \frac{u_k u_k^t}{u_k^t p_k} + \rho (p_k^t B_k p_k) v_k v_k^t \\ \text{où} \qquad v_k &= \frac{u_k}{u_k^t p_k} - \frac{B_k u_k}{u_k^t B_k u_k}. \end{aligned}$$

Le choix $\rho = 1$ a été proposé par Broyden, Fletcher, Goldfarb et Shanno (**BFGS**) alors que le choix $\rho = 0$ a été proposé par Davidon, Fletcher et Powell (**DFP**). Chaque technique possède des avantages et des inconvénients.

20. **rappel de programmation linéaire.** (voir également les notes du cours IFT 1571: marcotte/Ift1571/ift1571.html)

$$\boxed{\begin{array}{lllll} \min\limits_x & cx & & & \\ & Ax = b & c: & 1 \times n & \\ & x \geq 0 & x: & n \times 1 & \\ & & A: & m \times n & \\ & & b: & m \times 1 & \end{array}} \qquad (\mathrm{P})$$

On suppose que la matrice A (dont la $j$ième colonne sera notée A_j) est de plein rang. Soit B une sous-matrice carrée (**base**) formée de m colonnes indépendantes de A. On associe à B les décompositions $A = [B|N]$ et $x = \begin{pmatrix} x_B \\ x_N \end{pmatrix}$. Si l'on fixe le vecteur des **variables hors base** x_N à zéro, on obtient la **solution de base** $x = \begin{pmatrix} B^{-1}b \\ 0 \end{pmatrix}$. Une solution de base est **admissible** si $x_B \geq 0$. Réécrivons le programme linéaire (P) en mettant en évidence la décomposition de A et de x:

$$\begin{array}{ll} \min\limits_{x_B, x_N} & c_B x_B + c_N x_N \\ & B x_B + N x_N = b \\ & x_B, x_N \geq 0 \end{array}$$

ou, après quelques manipulations élémentaires:

$$\begin{array}{ll} \min\limits_{x_N} & c_B B^{-1} b + (c_N - c_B B^{-1} N) x_N \\ & B^{-1} b - B^{-1} N x_N \geq 0. \end{array}$$

Puisque $x_B = B^{-1}b \geq 0$, la solution $x_N = 0$ du programme ci-dessus est réalisable.

L'**algorithme du simplexe** tente d'améliorer la solution en intégrant dans la base une **variable hors-base** x_{j^*} ($j^* \in N$) dont le **coût réduit** $c_{j^*} - c_B B^{-1} A_{j^*}$ est négatif. La valeur maximale que peut prendre x_{j^*} est donnée par

$$x_{j^*} = \arg\inf_t \{t : (B^{-1}b)_i - t(B^{-1}A_{j^*})_i \geq 0 \ \forall i \in B\}.$$

Si $t = +\infty$, le programme linéaire (P) est non borné inférieurement. Sinon, on fait sortir de la base une variable x_{i^*} ($i^* \in B$) telle que

$$(B^{-1}b)_{i^*} - x_{j^*}(B^{-1}A_{j^*})_{i^*} = 0.$$

La nouvelle base est: $B \leftarrow B \cup A_{j^*} - A_{i^*}$.

On continue le processus jusqu'à ce que le vecteur des coûts réduits $c - c_B B^{-1} A$ soit non négatif. [**remarque:** la partie de ce vecteur correspondant aux variables de base est nulle puisque $c_B - c_B B^{-1} B = 0$.]

Dualité: Soit y un vecteur $1 \times m$. Une borne *inférieure* sur la valeur du programme (P) s'obtient en substituant aux contraintes $Ax = b$ l'unique contrainte $yAx = yb$. Si de plus on a que $yA \leq c$, on obtient trivialement la borne inférieure $cx \geq yAx = yb$. Si l'on veut maintenant obtenir la *meilleure* borne inférieure à partir de cette technique, on est amené à étudier le **programme linéaire dual**

$$\boxed{\begin{array}{ll} \max\limits_y & yb \\ & yA \leq c \end{array}} \qquad (\mathrm{D})$$

Remarquons que le dual du programme linéaire (D) est (P). De plus, si le problème primal (P) possède un minimum, on a le résultat:

$$\max_y yb = \min_x cx.$$

Les conditions **primales-duales** suivantes caractérisent l'optimalité d'une solution:

$$\left.\begin{aligned} Ax &= b \\ x &\geq 0 \end{aligned}\right\} \quad \text{primal réalisable}$$

$$yA \leq c \;\Big\} \quad \text{dual réalisable}$$

$$\left.\begin{aligned} yb &= cx \\ \text{ou } (yA - c)x &= 0 \end{aligned}\right\} \quad \begin{array}{l}\text{écarts complémentaires} \\ \text{(orthogonalité)}\end{array}$$

Les variables y sont appelées **variables duales** ou parfois **multiplicateurs du simplexe**. Si B est une base optimale, c'est-à-dire que les coûts réduits sont tous positifs ou nuls, alors $y = c_B B^{-1}$ est une solution optimale du problème dual.

On dit qu'une base (ou la solution de base correpondante) est **dégénérée** si au moins une des variables de base est nulle. La dégénérescence cause des problèmes dans l'analyse de la convergence de la méthode du simplexe.

Enfin, mentionnons qu'il existe des algorithmes polynomiaux pour la programmation linéaire, basés sur des techniques de programmation *non linéaire*. En toute probabilité, l'algorithme du simplexe n'est *pas* polynomial.

N.B. Dans ce qui suit, le gradient sera parfois un vecteur ligne, parfois un vecteur colonne, afin d'éviter la lourde notation $(\nabla f(x))^t$. Cela ne devrait pas créer de confusion.

1. **conditions nécessaires d'optimalité**: Considérons le programme

$$\begin{array}{ll} \min\limits_{x} & f(x) \\ & x \in X = \{x : Ax \geq b\} \end{array}$$

Si x est un candidat à l'optimalité (point stationnaire), il ne doit pas y avoir de direction de descente réalisable en x. Puisque toute direction réalisable peut prendre la forme $d = x' - x$ pour un certain $x' \in X$, on obtient la condition nécessaire d'optimalité du premier ordre:

$$\boxed{\nabla f(x)(x' - x) \geq 0 \qquad \forall x' \in X}$$

qu'on peut réécrire comme $\min_{x' \in X} \nabla f(x)x' = \nabla f(x)x$, c'est-à-dire :

$$x \in \arg\min_{x' \in X} \nabla f(x)x'.$$

Les conditions d'optimalité primales-duales de ce programme *linéaire* sont:

$$\boxed{\begin{array}{ll} Ax \geq b & \text{primal réalisable} \\ & \\ yA = \nabla f(x) & \text{dual réalisable} \\ y \geq 0 & \\ & \\ y(Ax - b) = 0 & \text{orthogonalité} \end{array}}$$

Ces conditions sont appelées conditions de **Karush-Kuhn-Tucker** associées à notre programme non linéaire. Ces conditions sont suffisantes si la fonction f est convexe. Si l'ensemble $X = \{x : g_1(x) \geq 0, \dots, g_m(x) \geq 0\}$ sont non linéaires, on obtient un résultat semblable en linéarisant les contraintes (moyennant certaines conditions techniques):

$$\boxed{\begin{array}{ll} g(x) \geq 0 & \text{primal réalisable} \\ & \\ y\nabla g(x) = \nabla f(x) & \text{dual réalisable} \\ y \geq 0 & \\ & \\ y\, g(x) = 0 & \text{orthogonalité} \end{array}} \qquad \text{KKT}$$

2. Une **méthode primale** engendre une suite de points satisfaisant les contraintes. La plupart de ces méthodes sont également des algorithmes de descente. Les algorithmes suivants seront exposés dans le contexte d'un programme non linéaire avec contraintes linéaires. Ils peuvent néanmoins être généralisés pour traiter des problèmes non linéaires plus généraux.

3. **Zoutendijk** (algorithme non fermé, convergence sous-linéaire): Soit $A_E x = b_E$ l'ensemble des contraintes satisfaites à égalité. Une direction de descente d^+ est obtenue en résolvant le sous-problème

$$\begin{array}{ll} \min\limits_{d} & \nabla f(x)d \\ & A_E d \geq 0 \\ & \|d\| \leq 1 \end{array}$$

$$\min_{\alpha \geq 0} \quad f(x + \alpha d^+) \qquad \rightarrow \qquad x^+ \quad .$$

Si $\|\cdot\| = \|\cdot\|_\infty$ ou $\|\cdot\| = \|\cdot\|_0$, la direction d^+ s'obtient en résolvant un programme linéaire.

4. **Frank-Wolfe** (algorithme fermé, convergence sous-linéaire): Soit y^+ la solution du programme linéaire

$$\begin{array}{ll} \min\limits_{y:Ay\geq b} & \nabla f(x)(y-x) \\ \min\limits_{0\leq\alpha\leq 1} & f(x+\alpha(y^+-x)) \quad \rightarrow \quad x^+. \end{array}$$

Cet algorithme est utile pour résoudre des problèmes de grande taille très structurés, problèmes de réseau par exemple.

5. **gradient projeté I** (algorithme fermé):

Soit γ positif et $p_\gamma = \mathrm{Proj}_X(x-\gamma\nabla f(x))$. On a $\min_{0\leq\alpha\leq 1} f(x+\alpha(p_\gamma - x)) \rightarrow x^+$. La direction $p_\gamma - x$ est solution optimale du problème quadratique

$$\min_{p\in X} \frac{1}{2}\|p-(x-\gamma\nabla f(x))\|^2$$

dont les conditions du premier ordre sont:

$$\langle p_\gamma - (x-\gamma\nabla(x)), p_\gamma - y\rangle \leq 0 \qquad \forall y \in X,$$

d'où l'on déduit, après avoir posé $y = x$ ci-dessus, que

$$\gamma\nabla f(x)p_\gamma \leq -\|p_\gamma - x\|^2,$$

c'est-à-dire que la direction est bien une direction de descente. Si f est fortement convexe (de module β) et son gradient lipschitzien (de constante L), la convergence est linéaire, même en prenant un pas de 1. En effet, puisque l'opérateur de projection est contractant et que $\mathrm{Proj}_X(x^* - \gamma\nabla f(x^*)) = x^*$, on peut écrire:

$$\begin{array}{rcl} \|p_\gamma - x^*\|^2 & \leq & \|x-\gamma\nabla f(x) - (x^* - \gamma\nabla f(x^*))\|^2 \\ & \leq & \|x-x^*\|^2 - 2\gamma(\nabla f(x) - \nabla f(x^*))(x-x^*) + \gamma^2\|\nabla f(x) - \nabla f(x^*)\|^2 \\ & \leq & (1-2\gamma\beta+\gamma^2L^2)\|x-x^*\|^2 \\ & < & \|x-x^*\|^2 \qquad \text{si} \quad \gamma < 2\beta/L^2. \end{array}$$

6. **gradient projeté II** (algorithme non fermé, convergence linéaire, si on est chanceux...):

Soit E l'ensemble des contraintes satisfaites à l'égalité et $d = \mathrm{Proj}_{\{p:A_Ep=0\}}(-\nabla f(x))$. On a

$$\min_{0\leq\alpha\leq 1} f(x+\alpha d) \quad \rightarrow \quad x^+.$$

La direction d est solution du programme quadratique:

$$\min_{A_Ed=0} \frac{1}{2}\|d-(-\nabla f(x))\|^2,$$

dont les conditions d'optimalité de Kuhn-Tucker sont:

$$\begin{array}{rcl} A_E^t y & = & d + \nabla f(x) \\ A_E d & = & 0. \end{array}$$

On suppose que la matrice A_E est de plein rang, ce qui permet, après avoir multiplié la première égalité par A_E, de trouver $y = (A_EA_E^t)^{-1}A_E\nabla f(x)$ et

$$d = -[I - A_E^t(A_EA_E^t)^{-1}A_E]\nabla f(x).$$

Si $d = 0$, on a $yA_E = \nabla f(x)$ et, si le vecteur de multiplicateurs y ne contient aucune composante négative, les conditions de Kuhn-Tucker sont satisfaites et la solution est optimale. Sinon, on recalcule la projection en supprimant une contrainte associée au multiplicateur de “mauvais” signe. Soit j l'indice d'une contrainte de multiplicateur négatif et $\overline{A}$ la matrice obtenue de A_E en supprimant la ligne A_j. Soit $\overline{d}$ la projection sur l'ensemble de contraintes réduit. On a:

$$\begin{array}{rcl} \nabla f(x) & = & A_E^t y \\ \nabla f(x) & = & \overline{A}^t\overline{y} - \overline{d}. \end{array}$$

Le vecteur $\overline{d}$ ne peut être nul, car alors le système $\nabla f(x) = A_E^t y$ aurait deux solutions distinctes, ce qui est incompatible avec l'hypothèse que la matrice A_E est de plein rang. De plus:

$$0 > \nabla f(x)\overline{d} = y^t A\overline{d} = y_j A_j \overline{d}.$$

Puisque y_j est négatif, on en déduit que $A_j\overline{d} > 0$ et que la direction $\overline{d}$ est admissible pour la ji'eme contrainte. Notons qu'en supprimant *simultanément*de A_E les lignes correspondant à plusieurs multiplicateurs négatifs, on n'est pas assuré que la direction résultante soit admissible.

7. **méthode des contraintes actives** (algorithme non fermé, convergence sous-linéaire):

   A chaque itération, on relaxe les contraintes qui ne sont pas actives, c'est-à-dire dont la variable duale est nulle. On se ramène ainsi à une optimisation en présence de contraintes d'égalité seulement, ce qui est plus simple. La difficulté consiste à gérer la liste des contraintes actives et inactives, à l'aide des variables duales.

8. **méthode du gradient réduit** (algorithme non fermé): Comme pour l'algorithme du simplexe, on utilise une décomposition du vecteur x en variables de base x_B et variables hors base x_N:

$$\begin{aligned} \min_{x_B, x_N} \quad & f(x_B, x_N) \\ & Bx_B + Nx_N = b \\ & x_B, x_N \geq 0. \end{aligned}$$

En posant $x_B = B^{-1}b - B^{-1}Nx_N$ on obtient:

$$\begin{aligned} \min_{x_N \geq 0} g(x_N) \quad &= \quad f(B^{-1}b - B^{-1}Nx_N, x_N) \\ \text{sujet à} \quad & \quad\; x_B = B^{-1}b - B^{-1}Nx_N \geq 0. \end{aligned}$$

Le **gradient réduit** $r(x)$ correspond au vecteur des coûts réduits en programmation linéaire:

$$r(x) = \nabla f(x) - \nabla_B f(x) B^{-1} A.$$

On a: $r_N(x) = \nabla g(x_N) = \nabla_N f(x) - \nabla_B f(x) B^{-1} N$ et $r_B(x) = 0$. Si $r(x) = 0$, la solution x satisfait les conditions d'optimalité du premier ordre. Sinon, on obtient une direction de descente:

$$\begin{array}{ll} j \in N: & d_j(x) = -r_j(x) \qquad \text{si } r_j(x) < 0 \text{ ou } x_j > 0; \\ j \in B: & d_B = -B^{-1}Nd_N, \end{array}$$

et on effectue la minimisation

$$\min_{\alpha \geq 0} f(x + \alpha d) \to x^+,$$

en s'assurant que le vecteur x demeure non négatif.

Comme dans l'algorithme du simplexe, toute variable de base qui s'annule quitte la base pour être remplacée par une variable hors base. Le choix de la variable hors base n'est pas toujours unique; en effet, même si la solution de base courante n'est pas dégénérée, il se peut qu'il y ait plusieurs variables hors base positives (ces variables sont appelées **superbasiques**). Il se peut aussi que le minimum de f dans la direction d soit atteint sans qu'aucune variable de base ne s'annule. On répète alors le processus sans changer de base.

La direction d ci-dessus n'est pas fermée. On peut cependant la rendre fermée en adoptant la modification suivante:

$$\begin{array}{lll} j \in N: & d_j(x) = -r_j(x) & \text{si } r_j(x) < 0; \\ & d_j(x) = -x_j r_j(x) & \text{si } r_j(x) > 0; \\ & & \\ j \in B: & d_B = -B^{-1}Nd_N. & \end{array}$$

On appelle algorithme du **simplexe convexe** la variante de l'algorithme du gradient réduit qui consiste à ne modifier qu'une seule variable hors base à chaque itération.

1. **méthodes de pénalités**: soit le problème générique

$$\min_{x \in S} \quad f(x)$$
$$S = \{x : g_i(x) \leq 0 \quad i = 1, \ldots, n\}.$$

On définit: $g_i^+(x) = \max\{0, g_i(x)\}$. La fonction P est une **fonction de pénalité** pour le problème ci-dessus si

$$\begin{array}{ll} P(x) = 0 & \forall x \in S \\ P(x) > 0 & \forall x \notin S. \end{array}$$

Cette fonction sert à définir le **problème pénalisé** (sans contraintes)

$$\boxed{\min_x f(x) + cP(x)} \qquad P(c)$$

dont on notera la solution optimale, qui dépend du **facteur de pénalité** c, par $x(c)$. Sous certaines conditions faibles, on a:

$$\lim_{c \to \infty} x(c) = x^*.$$

Les formes les plus classiques des fonctions de pénalités sont:

$$P(x) = \tfrac{1}{2} \textstyle\sum_{i=1}^n [\max\{0, g_i(x)\}]^2 = \tfrac{1}{2} \|g^+(x)\|_2^2$$

$$P(x) = \textstyle\sum_{i=1}^n \max\{0, g_i(x)\} = \|g^+(x)\|_\infty.$$

Soit $P(x) = \frac{1}{2}\|g^+(x)\|_2^2$. A l'optimum de $P(c)$ on a:

$$\nabla f(x(c)) + cg^+(x(c))\nabla g(x(c)) = 0.$$

Posons $y(c) = cg^+(x(c))$. Si $x(c) \to x^*$ et que le point x^* est un point régulier (cette condition est satisfaite si les gradients des contraintes actives en x^* sont linéairement indépendants), on a: $\lim_{c \to \infty} y(c) = y^*$, le vecteur dual optimal correspondant à la solution primale optimale x^*.

Sous certaines conditions de régularité, la fonction de pénalité $P(x) = \|g^+(x)\|_\infty$ est **exacte**, c'est-à-dire qu'il existe une valeur critique c^* *finie* telle que la solution de $P(c)$ soit x^* pour tout $c \geq c^*$. Notons cependant que le problème pénalisé est alors non différentiable.

Les sous-problèmes sans contraintes sont résolus par des méthodes efficaces: Newton, quasi-Newton, gradients conjugués, etc.

2. Une **fonction de barrière** B satisfait les deux conditions

$$\begin{array}{ll} B(x) \geq -\infty & \text{si } x \in S \\ B(x) \to \infty & \text{lorsque } x \text{ approche la frontière de } S \end{array}$$

et sert à définir le problème

$$\boxed{\min_x f(x) + \tfrac{1}{c}B(x)} \qquad Q(c)$$

dont on notera la solution optimale, qui dépend du **facteur de barrière** c, par $x(c)$. Sous certaines conditions faibles, on a:

$$\lim_{c \to \infty} x(c) = x^*.$$

Deux formes classiques de fonction de barrière sont:

$$B(x) = -\textstyle\sum_{i=1}^p \frac{1}{g_i(x))}$$

$$B(x) = -\textstyle\sum_{i=1}^p \ln(g_i(x)) \quad \text{(barrière logarithmique)}$$

Si B est de la première forme on a, à l'optimum de $Q(c)$:

$$\nabla f(x(c)) + \sum_{i=1}^{p} \frac{1}{c[g_i(x(c))]^2} \nabla g_i(x(c)) = 0.$$

Posons $y(c) = 1/c[g_i(x(c))]^2$. On a, sous certaines conditions: $\lim_{c\to\infty} y(c) = y^*$.

Les problèmes sans contraintes sont résolus à l'aide d'algorithmes efficaces.

3. Une **méthode primale-duale** pour résoudre le problème *convexe*

$$\begin{array}{ll} \min_x & f(x) \\ \text{sujet à} & g(x) \leq 0. \end{array}$$

Définissons le **lagrangien**

$$L(x, y) = f(x) + yg(x).$$

L'optimum primal-dual (x^*, y^*) est un **point de selle** du lagrangien, c'est-à-dire :

$$f(x^*) + yg(x^*) \leq f(x^*) + y^*g(x^*) \leq f(x) + y^*g(x) \qquad \forall x, y.$$

On définit la **fonction duale**

$$\phi(y) = \min_x L(x, y).$$

La fonction ϕ est concave et son maximum est atteint en y^*. L'algorithme de Arrow consiste à maximiser ϕ à l'aide de la méthode du gradient projeté I avec pas fixe. Si la fonction f est strictement convexe, le gradient de ϕ est simplement $g(x)$, où x est l'unique minimum de la fonction lagrangienne pour y fixé. On peut également utiliser une méthode primale duale:

$$\begin{array}{rcl} x^+ & = & x - \alpha[\nabla f(x) + y\nabla g(x)]^t \\ y^+ & = & \max\{0, y + \alpha g(x)\}, \end{array}$$

connue sous le nom d'algorithme de Arrow-Hurwicz-Uzawa, et qui converge linéairement si la fonction f est fortement convexe.

4. La **méthode des multiplicateurs** est un compromis entre la méthode primale-duale de la section précédente et la méthode des pénalités. Considérons le problème

$$\min_{x:h(x)=0} f(x) \equiv \min_{x:h(x)=0} f(x) + \frac{1}{2}c\|h(x)\|_2^2$$

et son lagrangien augmenté $L(x, y, c) = f(x) + yh(x) + \frac{1}{2}c\|h(x)\|_2^2$. On minimise le lagrangien successivement par rapport aux variables x et y, ce qui donne:

$$\begin{array}{l} \min\limits_x L(x, y, c) \to x^+ \\ y^+ = y + ch(x). \end{array}$$

On peut montrer qu'il existe une valeur *finie* c^* telle que $L(x, y, c)$ possède un minimum local en x^* pour tout $c \geq c^*$. De plus, sous des conditions faibles, la convergence est linéaire.

Si le problème comporte des contraintes d'inégalité, on transforme celles-ci en contraintes d'égalité en introduisant des variables d'écart. Ainsi, on remplace la contrainte $g_i(x) \leq 0$ par la contrainte $g_i(x) + z_i^2 = 0$.

1. **méthodes des ellipsoïdes**

   Cette méthode permet de trouver un point réalisable, s'il en existe un, pour un ensemble X défini à partir de fonctions convexes: $X = \{x : g_i(x) < 0, i \in I\}$. Soit M_0 une matrice symétrique définie positive telle que $X \subset E_0 = \{x : x^t M_0^{-1} x \leq 1\}$. Soit x^0 le centre de l'ellipsoïde E_0. Si $x^0 \notin X$, soit i l'indice d'une contrainte violée. Puisque g_i est convexe, on a:

   $$X \subset \{x : g_i(x^0) + \nabla g_i(x^0)(x - x^0) \leq 0\}.$$

   L'algorithme construit l'ellipsoïde de volume minimal contenant l'intersection de E_0 avec le demi-espace défini par l'inéquation précédente, et répète le processus jusqu'à ce que soit le centre du nouvel ellipsoïde fasse partie de X, soit l'on puisse démontrer que X est vide.

   Si l'**oracle** qui produit l'indice d'une contrainte violée est polynomial, alors l'algorithme est polynomial. C'est le cas pour la programmation linéaire (Khachyan).

2. **méthode primale-duale**

   Considérons le programme mathématique

   $$\begin{aligned} \max_x \quad & f(x) \\ & g_i(x) = 0 \quad i \in I \\ & x \geq 0 \end{aligned}$$

   et ses conditions d'optimalité:

   $$\boxed{\begin{array}{ll} g_i(x) = 0 & i \in I \\ \nabla f(x) - \sum_{i \in I} y_i \nabla g_i(x) = s & \\ s_j x_j = 0 & j \in [1..n] \end{array}} \qquad \text{KKT}$$

   Considérons maintenant une méthode de barrière pour la résolution de ce programme. Pour un paramètre μ donné, on obtient le programme mathématique (dont la solution est unique)

   $$\begin{aligned} \min_x \quad & f(x) - \mu \sum_j \log x_j \\ & g_i(x) = 0 \quad i \in I \end{aligned}$$

   dont les conditions d'optimalité sont

   $$\begin{aligned} g_i(x) &= 0 \quad i \in I \\ \nabla f(x) - \mu(1/x_1, \ldots, 1/x_n) - \sum_{i \in I} y_i \nabla g_i(x) &= 0. \end{aligned}$$

   En posant $s_j = \mu / x_j$, on peut réécrire ce système comme un système de Karush-Kuhn-Tucker perturbé:

   $$\boxed{\begin{array}{ll} g_i(x) = 0 & i \in I \\ \nabla f(x) - \sum_{i \in I} y_i \nabla g_i(x) = s & \\ s_j x_j = \mu & j \in [1..n] \end{array}} \qquad \text{KKT}(\mu)$$

L'idée générique d'une classe de méthodes de points intérieurs est de résoudre, à l'aide de la méthode de Newton, le système de Kuhn-Tucker initial ou le système KT(μ) en s'assurant, dans ce dernier cas, que le paramètre μ ne décroisse pas trop rapidement d'une itération à l'autre, afin que les conditions de non négativité soient toujours satisfaites et qu'une itération de la méthode de Newton nous rapproche de la solution du système de Kuhn-Tucker initial.

La courbe paramétrée $x(\mu)$ est appelée **chemin central** et l'algorithme ne s'en éloigne jamais "trop". Si la solution n'est pas unique, l'algorithme produit une suite qui converge vers le **centre analytique** de l'ensemble des solutions.

3. **cas linéaire (méthode de chemin central)**

Dans le cas linéaire, on fait l'hypothèse que la matrice des contraintes A est de plein rang; le système KT(μ) prend la forme

$$\begin{aligned} Ax &= b \\ A^t y + s &= c \\ Xs &= \mu e \qquad\qquad (X = \mathrm{diag}(x_1, \ldots, x_n), \quad e = (1, \ldots, 1)^t) \end{aligned}$$

et la direction de Newton est solution du système linéaire

$$\begin{pmatrix} A & 0 & 0 \\ 0 & A^t & I \\ S & 0 & X \end{pmatrix} \begin{pmatrix} d_x \\ d_y \\ d_s \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \mu e - XSe \end{pmatrix} \qquad\qquad (S = \mathrm{diag}(s_1, \ldots, s_n))$$

La solution de ce système est

$$\begin{pmatrix} d_y \\ d_s \\ d_x \end{pmatrix} = \begin{pmatrix} -(AS^{-1}XA^t)^{-1}AS^{-1}(\mu e - XSe) \\ -A^t d_y \\ S^{-1}(\mu e - XSe) - S^{-1}X d_s \end{pmatrix}.$$

Si μ ne diminue pas trop rapidement d'une itération à l'autre, on peu effectuer un pas unitaire. Ceci sera le cas si le facteur de réduction est fixé à $1-\beta n^{-1/2}$, avec $\beta \leq 1/5$. Des méthodes plus "agressives" utilisent des réductions plus fortes, compensées par un plus grand nombre d'itérations de la méthode de Newton. Ces dernières accélèrent la convergence. Dans les deux cas, on peut démontrer la polynomialité de l'algorithme.

Note: puisque $y = (AA^t)^{-1}A(c-s)$, on peut se contenter d'appliquer l'algorithme dans l'espace primal des vecteurs x et s.

4. **cas linéaire (méthode primale affine)**

Soit le programme linéaire

$$\min_{Ax=b, x\geq 0} cx$$

et soit $\overline{x}$ un point admissible dont toutes les composantes soient positives. En effectuant le changement de variable $u = X^{-1}x$, on obtient le programme linéaire

$$\begin{aligned} \min \qquad & cXu \\ & AXu = b \\ & u \geq 0. \end{aligned}$$

Au point $\overline{u} = X^{-1}\overline{x} = e$, la direction de plus forte descente est

$$d_u = \mathrm{proj}_{\{d: AXd=0\}}(-Xc) = -(I - XA^t(AX^2A^t)^{-1}AX)Xc$$

et l'on pose

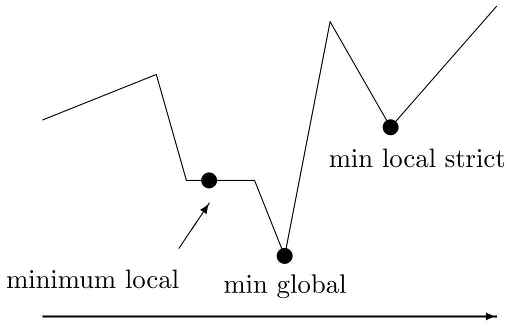
$$\begin{aligned} u^+ &= \overline{u} + \alpha d_u \\ x^+ &= \overline{x} + \alpha X d_u \\ &= \overline{x} - \alpha X(I - XA^t(AX^2A^t)^{-1}AX)Xc. \end{aligned}$$

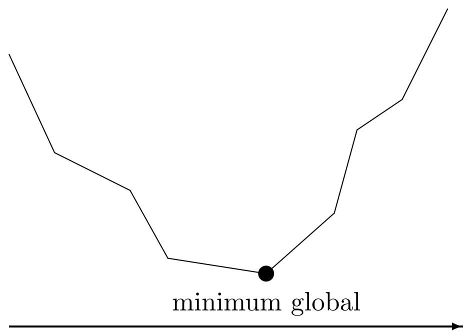
On pose $\alpha = \beta\alpha_{\max}$ où $\alpha_{\max}$ est le pas admissible maximal et $\beta \approx 1$ ($\beta < 1$). Sous cette forme simpliste, l'algorithme, quoique très performant en pratique, n'est pas polynomial.

TECHNIQUES D'OPTIMISATION: EXEMPLES

CHAPITRE 1: GÉNÉRALITÉS. OPTIMISATION SANS CONTRAINTES. SIMPLEXE

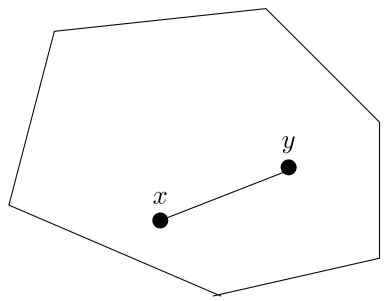
1.



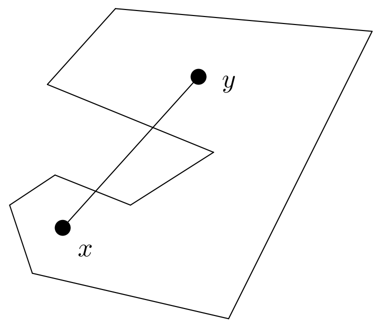


fonction unimodale

2.

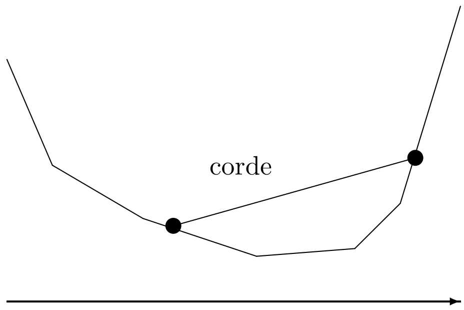


ensemble convexe (polytope)

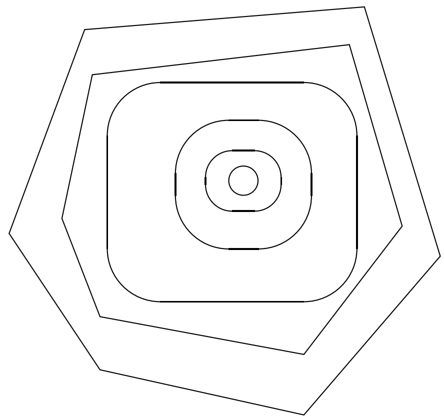


"polyèdre" non convexe

3. Une fonction linéaire est convexe mais pas strictement convexe. La fonction $f(x) = x^2$ est strictement convexe.



fonction convexe



courbes de niveau d'une fonction quasiconvexe

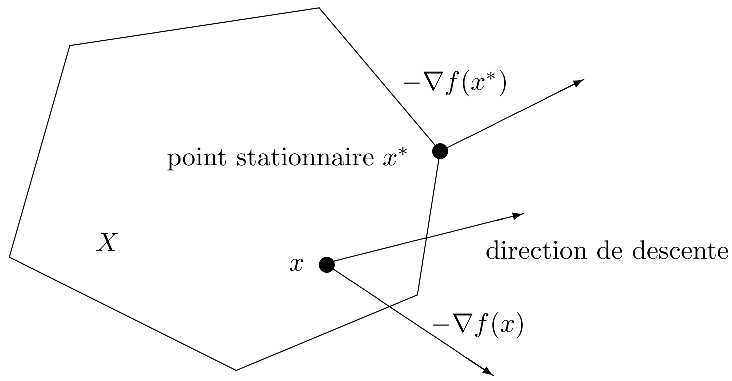
4.

$$f(x,y) = xy + x^2 \quad \nabla f(x,y) = (\, y+2x \quad x \,) \quad H(x,y) = \begin{pmatrix} 2 & 1 \\ 1 & 0 \end{pmatrix}$$

$$f(1,2) = 3 \qquad \nabla f(1,2) = (\, 4 \quad 1 \,) \qquad H(1,2) = \begin{pmatrix} 2 & 1 \\ 1 & 0 \end{pmatrix}$$

$$f(x,y) \approx 3 + (\, 4 \quad 1 \,) \begin{pmatrix} x-1 \\ y-2 \end{pmatrix} + \tfrac{1}{2} (\, x-1 \quad y-2 \,) \begin{pmatrix} 2 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x-1 \\ y-2 \end{pmatrix}$$

5.



Soit $f(x,y) = x^2y - y$. Les points stationnaires de f sont des zéros du gradient $\nabla f(x,y) = (\, 2xy \quad x^2-1 \,)$, c'est-à-dire $(\, \pm 1 \quad 0 \,)$. En ces deux point, la matrice hessienne vaut $H(\pm 1, 0) = \begin{pmatrix} 0 & \pm 2 \\ \pm 2 & 0 \end{pmatrix}$. Puisqu'elle est **indéfinie**, ces deux points ne sont ni des maximums ni des minimums.

6. Soit le programme $\min_{y \geq 0} f(x,y) = x^2 + y^3 + y$ pour lequel le point $(x^*, y^*) = (0,0)$ est stationnaire. On a:

$$\nabla f(x,y) = (\, 2x \quad 3y^2 \,) \qquad \nabla f(x^*,y^*) = (\, 0 \quad 0 \,)$$
$$H(x,y) = \begin{pmatrix} 2 & 0 \\ 0 & 6y \end{pmatrix} \qquad H(x^*,y^*) = \begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}.$$

Les directions d telles que $\nabla f(x^*,y^*)d = 0$ sont $d = (\pm 1, 0)$. Ces directions satisfont: $d^t H(x^*,y^*)d = 2 > 0$ et par conséquent le point (x^*, y^*) est un minimum local (un minimum global en fait).

7. L'algorithme du gradient (voir plus loin) est un algorithme de descente pour la fonction $Z(x) = f(x)$. De plus, il est fermé, si l'on définit $X = R^n$, $X^* = \{\text{points stationnaires}\} = \{x : \nabla f(x) = 0\}$, et que l'on effectue une recherche linéaire fermée à chaque itération, telle que spécifiée par la règle d'Armijo, par exemple. (voir plus loin.)

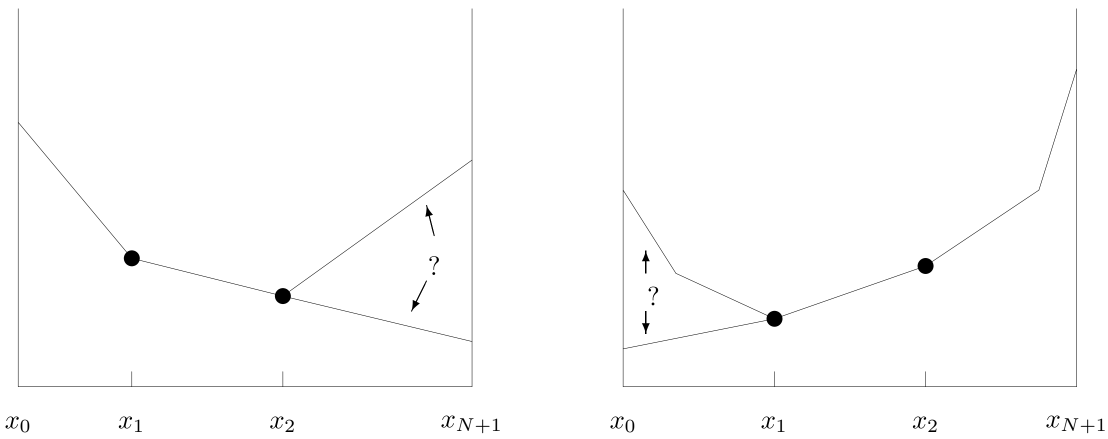
8. • Soit $\gamma \in (0,1)$ et $r_k = \gamma^k$. On a: $r^* = \lim_{k\to\infty} r_k = 0$ et: $\lim_{k\to\infty}(|r_{k+1} - r^*|/|r_k - r^*|) = \gamma$. La convergence est donc linéaire avec taux γ.

   • Soit $r_k = (1/k)^k$. On a: $\lim_{k\to\infty}(r_{k+1}/r_k^p) = \infty$ si $p > 1$. Par contre: $\lim_{k\to\infty}(r_{k+1}/r_k) = 0$. La convergence de la suite vers zéro est donc superlinéaire.

   • Soit $a \in (0,1)$ et $r_k = a^{2^k}$. Puisque $\lim_{k\to\infty}(r_{k+1}/r_k^2) = 1$, la convergence de la suite vers zéro est quadratique.

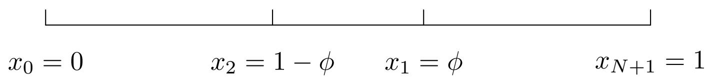
   • La suite $r_k = 1/k$ converge sous-linéairement vers zéro.

9. • La diminution de l'intervalle d'incertitude par l'évaluation d'un nouveau point:

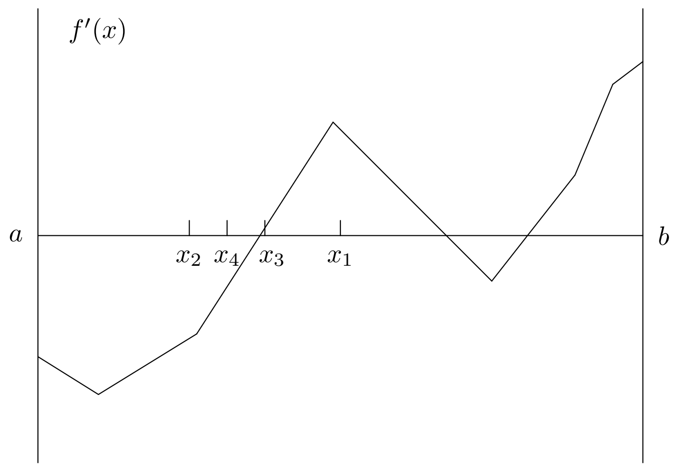


- Suite de points d'évaluation découlant de l'application de la méthode de Fibonacci, si la plus petite valeur observée de la fonction est toujours obtenue au dernier point d'évaluation.

- Les deux premiers points d'évaluation correspondant à la méthode de la section dorée.



10. Recherche dichotomique d'un zéro d'une fonction continue.



11. On peut approcher la valeur de $\sqrt{2}$ en appliquant la méthode de Newton à l'équation $x^2 - 2 = 0$. On obtient la suite définie par:

$$x^{k+1} = x^k - \frac{(x^k)^2 - 2}{2x^k} = \frac{x^k}{2} + \frac{1}{x^k}.$$

Si $x^0 = 1$ on obtient:

$$\begin{aligned} x^1 &= 1/2 + 1/1 = 1.5 \\ x^2 &= 3/4 + 2/3 = 1.416666667 \\ x^3 &= 1.414215686 \\ x^4 &= 1.414213562. \end{aligned}$$

12. Si on applique la méthode de la fausse position à la fonction $x^2 - 2$, on obtient:

$$x^{k+1} = x^k - ((x^k)^2 - 2)\frac{x^k - x^{k-1}}{(x^k)^2 - (x^{k-1})^2} = x^k - \frac{(x^k)^2 - 2}{x^k + x^{k-1}}.$$

Si $x^0 = 0$ et $x^1 = 1$ on obtient la suite:

$$\begin{array}{rcl} x^2 & = & 1 - (-1/1) = 2 \\ x^3 & = & 1.333333333 \\ x^4 & = & 1.400000000 \\ x^5 & = & 1.414634146 \\ x^6 & = & 1.414211439 \\ x^7 & = & 1.414213562 \end{array}$$

13.

14. • Soit $f(x) = \frac{1}{2}x^tQx + b^tx$. On a $\nabla^t f(x) = Qx + b$ et la méthode du gradient prend la forme: $x^{k+1} = x^k - \alpha_k(Qx^k + b)$. Soit $g^k = \nabla^t f(x^k)$ et $\phi(\alpha) = f(x^k - \alpha_k g^k)$. On veut:

$$\begin{array}{rcl} \phi'(\alpha_k) & = & \nabla f(x^k)\nabla^t f(x^{k+1}) = 0 \\ & \to & (Qx^k + b)^t(Q(x^k - \alpha_k g^k) + b) = 0 \\ & \to & \alpha_k = \dfrac{\|Qx^k + b\|^2}{(Qx^k + b)^tQ(Qx^k + b)}. \end{array}$$

• Si $Q = \begin{pmatrix} 3 & 1 \\ 1 & 4 \end{pmatrix}$ dans l'exemple ci-dessus, on obtient: $a = 2$ et $A = 5$. Le taux de convergence est donc, par l'inégalité de Kantorovitch, inférieur à $(A - a)^2/(A + a)^2 = (5 - 2)^2/(5 + 2)^2 = 9/49$.

15. Soit le système d'équations: $x^2 + y - 3 = 0$, $2x - y^2 + 2 = 0$. Au point de départ $x^0 = (1 \quad 1)^t$, on a: $f(x^0) = (-1 \quad 3)^t$. Il s'ensuit:

$$J(x, y) = \begin{pmatrix} 2x & 1 \\ 2 & -2y \end{pmatrix} \qquad J(1, 1) = \begin{pmatrix} 2 & 1 \\ 2 & -2 \end{pmatrix}$$

$$x^1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 2 & 1 \\ 2 & -2 \end{pmatrix}^{-1} = \frac{1}{6}\begin{pmatrix} -2 & -1 \\ -2 & 2 \end{pmatrix}\begin{pmatrix} -1 \\ 3 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix} - \begin{pmatrix} 1/6 \\ -4/3 \end{pmatrix} = \begin{pmatrix} 5/6 \\ 7/3 \end{pmatrix}.$$

16. Soit $f(x, y) = \frac{1}{2}(x + y)^2 + x^2 - y$. On a:

$$\begin{array}{rcl} \min\limits_x f(x, y) & \to & x^+ + y + 2x^+ = 0 \Rightarrow x^+ = -\dfrac{1}{3}y \\ \min\limits_y f(x, y) & \to & x^+ + y^+ - 1 = 0 \to y^+ = 1 - x^+ = 1 + \dfrac{1}{3}y^+. \end{array}$$

A partir de $x^0 = 1$ et $y^0 = 2$, l'algorithme de Gauss-Seidel produit la suite:

$$\begin{array}{ll} x^1 = -2/3 & y^1 = 5/3 \\ x^2 = -5/9 & y^2 = 14/9 \\ x^3 = -14/27 & y^3 = 41/27 \\ x^4 = -41/81 & y^4 = 122/81 \,. \end{array}$$

17. Soit $\phi(\alpha) = (\alpha - 1)^2$. On a: $\phi(0) = 1$ et $\phi'(0) = -2$.

• Armijo ($\epsilon = 1/2$ et $\eta = 2$):

$$\begin{array}{rcl} (\alpha - 1)^2 & \leq & 1 + \dfrac{1}{2}(-2)\alpha = 1 - \alpha \Rightarrow \alpha \leq 1 \\ (2\alpha - 1)^2 & > & 1 + \dfrac{1}{2}(-2)2\alpha = 1 - 2\alpha \Rightarrow \alpha > \dfrac{1}{2} \end{array}$$

- Goldstein ($\epsilon = 1/3$):

$$
\begin{aligned}
(\alpha-1)^2 &\leq 1-\frac{2}{3}\alpha \;\Rightarrow\; \alpha \leq \frac{4}{3}\\
(\alpha-1)^2 &> 1-\frac{4}{3}\alpha \;\Rightarrow\; \alpha > \frac{2}{3}
\end{aligned}
$$

- Wolfe ($\epsilon = 1/3$): $\alpha \leq 4/3$ comme pour Goldstein et:

$$2(\alpha-1) > \frac{2}{3}(-2) \;\rightarrow\; \alpha > 1-\frac{2}{3} = \frac{1}{3}.$$

18.

$$\min\; 2x_1^2 + x_1x_2 + x_2^2 - 6x_1 - 5x_2 \qquad g(x) = \begin{pmatrix} 4x_1+x_2-6 \\ x_1+2x_2-5 \end{pmatrix} \qquad Q = \begin{pmatrix} 4 & 1 \\ 1 & 2 \end{pmatrix} \qquad x^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

$k=0 \quad p_0 = -g_0 = \begin{pmatrix} 6 \\ 5 \end{pmatrix}$

$$
\begin{aligned}
&\alpha_0 = \|g_0\|^2/p_0^tQp_0 = (6^2+5^2)/(6\;5)Q(6\;5)^t = 61/254\\
&x^1 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} + 61/254 \begin{pmatrix} 6 \\ 5 \end{pmatrix} \approx \begin{pmatrix} 1.44 \\ 1.20 \end{pmatrix}\\
&g_1 \approx \begin{pmatrix} 4\times 1.44 + 1.20 - 6 \\ 1.44 + 2\times 1.20 - 5 \end{pmatrix} \approx \begin{pmatrix} .96 \\ -1.16 \end{pmatrix}\\
&\beta_0 = \|g_1\|^2/\|g_0\|^2 \approx 2.27/61 \approx .037 \qquad \text{(Fletcher-Reeves)}\\
&\beta_0 = (g_1-g_0)^tg_1/\|g_0\|^2 \approx (-5.04\;\; -6.16)^t(.96\;\; -1.16)/61 \approx 2.31/61 \approx .038 \qquad \text{(Polak-Ribère)}
\end{aligned}
$$

$k=1 \quad p_1 = -g_1 + \beta_0 p_0 \approx \begin{pmatrix} -.96 \\ 1.16 \end{pmatrix} + .037 \begin{pmatrix} 6 \\ 5 \end{pmatrix} \approx \begin{pmatrix} -.74 \\ 1.35 \end{pmatrix}$

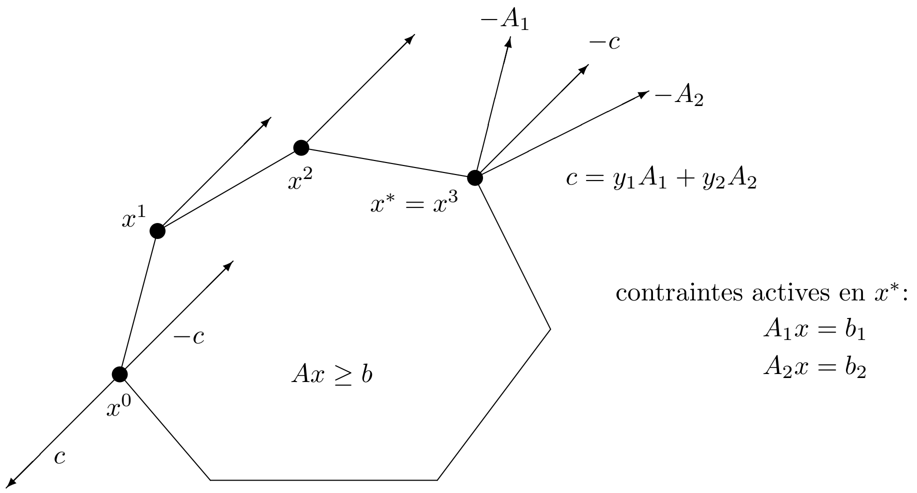
$$
\begin{aligned}
&\alpha_1 = \|g_1\|^2/p_1^tQp_1 = 2.27/3.84 \approx .59\\
&x^1 \approx \begin{pmatrix} 1.44 \\ 1.20 \end{pmatrix} + .59 \begin{pmatrix} -.74 \\ 1.35 \end{pmatrix} \approx \begin{pmatrix} 1.00 \\ 2.00 \end{pmatrix} \approx x^*.
\end{aligned}
$$

On vérifie que les directions p_0 et p_1 sont bien conjuguées:

$$p_1^tQp_0 \approx (-.74\;1.35)^t \begin{pmatrix} 4 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} 6 \\ 5 \end{pmatrix} \approx (-.74\;1.35) \begin{pmatrix} 29 \\ 16 \end{pmatrix} \approx -.14 \approx 0.$$

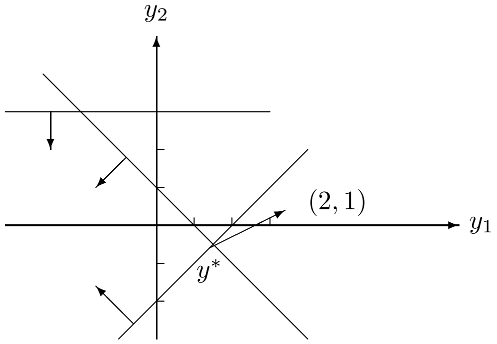
19. •

20. •



- Considérons le couple de programmes linéaires primal-dual

$$\begin{array}{lllll} \min & x_1+3x_2+2x_3 = 2 & \qquad & \max & 2y_1+y_2 \\ & x_1 \qquad - \ x_3 = 1 & & & y_1+y_2 \leq 1 \\ & x_1\ ,\ \ x_2\ ,\ \ x_3 \geq 0 & & & y_1-y_2 \leq 2 \end{array}$$



Puisque y_1^* et y_2^* son positifs, on en déduit que $x_1^* - x_3^* = 2$ et $x_1^* + x_2^* + x_3^* = 1$. Symétriquement, puisque $y_2 < 3$, il faut que x_2^* soit nul. Par conséquent: $x_1^* = x_2^* = 1/2$, et on vérifie aisément que les objectifs primal et dual sont égaux à 2.

1. •

$$\begin{array}{lcl} \min f(x) & = & 2x_1^2 + 2x_1x_2 + x_2^2 - 10x_1 - 10x_2 \\ \text{sujet à} & & x_1^2 + x_2^2 \leq 5 \\ & & 3x_1 + x_2 \leq 6 \end{array}$$

Les conditions de KKT du problème précédent s'expriment comme

$$\begin{array}{l} \begin{pmatrix} 4x_1 + 2x_2 - 10 \\ 2x_1 + 2x_2 - 10 \end{pmatrix} + y_1 \begin{pmatrix} 2x_1 \\ 2x_2 \end{pmatrix} + y_2 \begin{pmatrix} 3 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ y_1, y_2 \geq 0 \\ y_1(x_1^2 + x_2^2 - 5) = 0 \\ y_2(3x_1 + x_2 - 6) = 0. \end{array}$$

La solution de ce système est: $x^* = (1,2)^t$, $y^* = (1,0)$.

• Plusieurs situations font intervenir des contraintes linéaires:

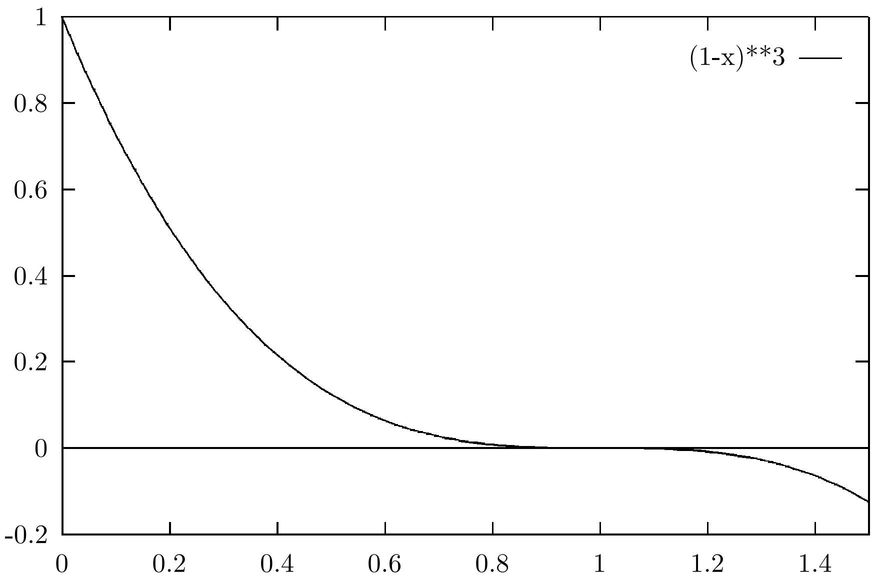
$$\begin{array}{ll} \min_{x \geq 0} & f(x) \\ & Ax = b. \end{array}$$

Les conditions de KKT pour de tels problèmes s'écrivent:

$$\boxed{\begin{array}{ll} Ax = b, \quad x \geq 0 & \text{primal réalisable} \\[1em] \nabla f(x) - yA \geq 0 & \text{dual réalisable} \\[1em] (\nabla f(x) - yA)x = 0 & \text{orthogonalité} \end{array}} \qquad \text{(KKT)}$$

2. L'optimum du problème

$$\begin{array}{ll} \min_{x,y \geq 0} & -x \\ \text{sujet à} & (1-x)^3 - y \geq 0 \end{array}$$

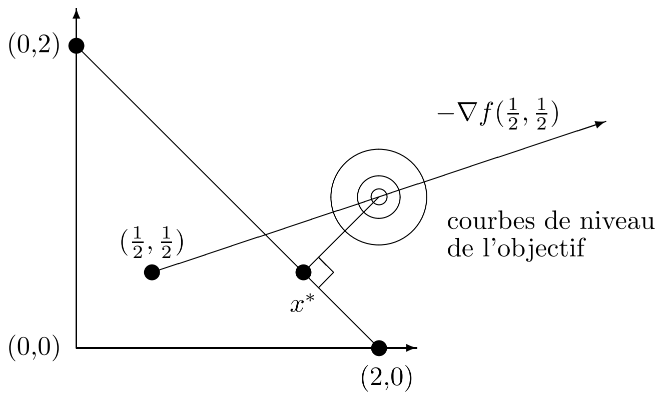


est le point $(1,0)^t$. Pourtant, en ce point, il n'existe pas (voir figure ci-dessous) de multiplicateur y satisfaisant la condition de KKT

$$\nabla f(1,0) - \lambda \nabla g(1,0) = (-1,0) - y(0,-1) \geq 0.$$

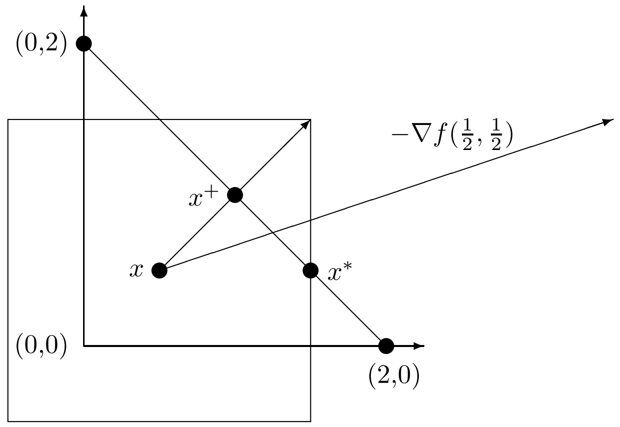
3. Considérons le problème quadratique convexe

$$\begin{array}{ll} \min_{x \geq 0} & (x_1-2)^2+(x_2-1)^2 \\ \text{sujet à} & x_1+x_2 \leq 2. \end{array}$$

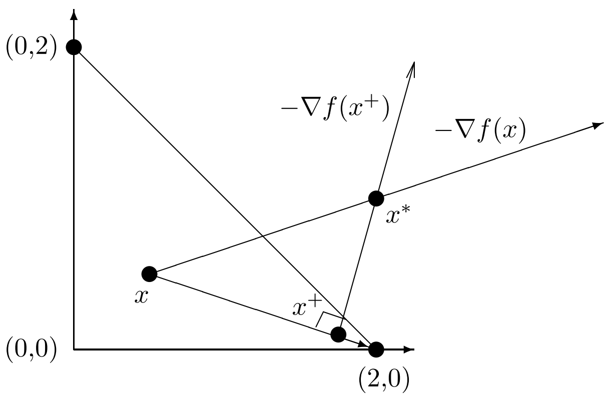


Les figures ci-dessous illustrent les directions de descente obtenues à partir de divers algorithmes, en prenant comme point de référence $(1/2, 1/2)$, dont le gradient est $(-3/2, -1/2)$.

Zoutendijk (norme $\|\cdot\|_\infty$)

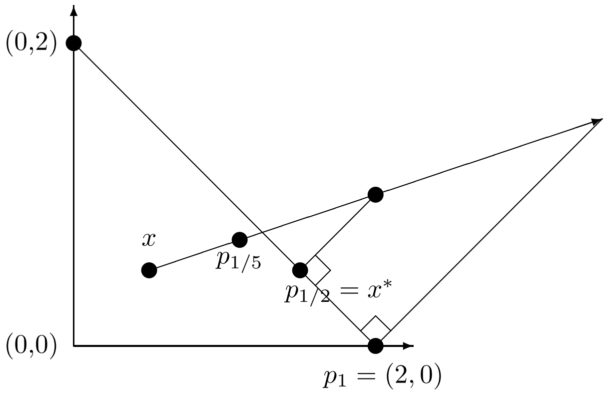


4. Frank-Wolfe

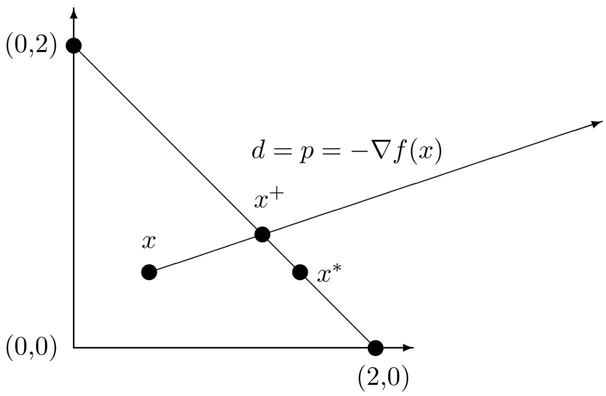


5.

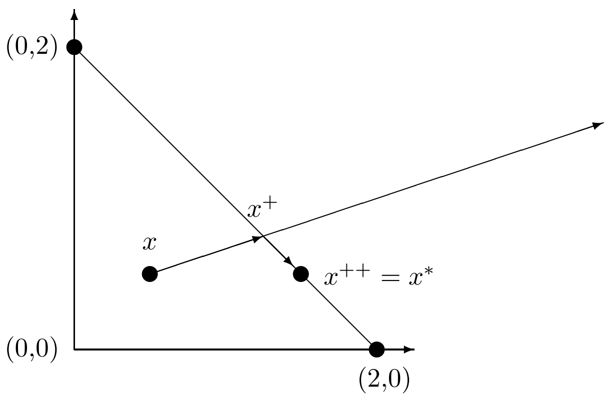
gradient projeté I



6. gradient projeté II



7. méthode des contraintes actives



8.

$$\begin{array}{lcl} \min_{x\geq 0} f(x) & = & x_1^2 + x_2^2 + x_3^2 + x_4^2 - 2x_1 - 3x_4 \\ \text{sujet à} & & 2x_1 + x_2 + x_3 + 4x_4 = 7 \\ & & x_1 + x_2 + 2x_3 + x_4 = 6 \end{array}$$

Soient $x = (2,2,1,0)^t$, x_1 et x_2 les variables de base et $B = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$ la base. On a: $\nabla f(x) = (2,4,2,-3)$. On a:

$$r(x) = (2,4,2,-3) - (2,4)\begin{pmatrix} 1 & -1 \\ -1 & 2 \end{pmatrix}\begin{pmatrix} 2 & 1 & 1 & 4 \\ 1 & 1 & 2 & 1 \end{pmatrix} = (0,0,-8,-1).$$

par conséquent, $d_3 = 8$ et $d_4 = 1$. On peut écrire:

$$\begin{aligned} x_1 &= 1 + x_3 - 3x_4 \\ x_2 &= 5 - 3x_3 + 2x_4 \end{aligned}$$

$$\begin{aligned} x_1^+ &= 1 + x_3 + 8\alpha - 3(x_4 + \alpha) = x_1 + 5\alpha \\ x_2^+ &= 5 - 3(x_3 + 8\alpha) + 2(x_4 + \alpha) = x_2 - 22\alpha \end{aligned}$$

d'où l'on tire que $d_1 = 5$, $d_2 = -22$ et $\alpha \leq 1/11$. On vérifie que le minimum de la fonction $f(x + \alpha d)$ pour $\alpha \in [0, 1/11]$ est obtenu pour $\alpha = 1/11$. La variable x_2 quitte alors la base pour y être remplacée par x_3 ou x_4, au choix.

On note que la direction du gradient réduit ne coïncide pas nécessairement avec la direction du gradient projeté sur le cône des des directions admissibles! En effet, le choix de la base peut influencer la direction du gradient réduit.