

LES METHODES DE POINTS INTERIEURS

1 INTRODUCTION:

- SIMPLEXE
 - COMMENT PASSER PAR L'INTERIEUR? DIFFICULTES AUX BORDS
- !!!
- ON VA VOIR 3 METHODES

1.1 LA METHODE “AFFINE SCALING”

- CERTAINEMENT LA PLUS SIMPLE DES METHODES DE POINTS INTERIEURS
 - ELLE COMBINE SIMPLICITE AVEC DE TRES BONNES PERFORMANCES EN PRATIQUE
 - ON A PU OBSERVER QUE, SI ON DEMARRE D'UN POINT PRES DE L'OPTIMUM, L'ALGO SE DEPLACE A PEU PRES LE LONG DES ARETES DU DOMAINE.
 - ELLE UTILISE L'IDEE D'OPTIMISER SUR UNE ELLIPSOÏDE

1.2 LE “POTENTIAL REDUCTION ALGORITHM”

ELLE INTRODUIT UNE DEUXIEME IDEE: AU LIEU DE MESURER LE PROGRES VERS L'OPTIMUM UNIQUEMENT EN TERME DE LA FONCTION ECONOMIQUE, ON MESURE CE PROGRES PAR LA DIMINUTION D'UNE CERTAINE FONCTION POTENTIELLE NON LINEAIRE
CETTE FONCTION ESSAIE D'EQUILIBRER DEUX OBJECTIFS A PRIORI CONTRADICTOIRS:

1. DIMINUER LA VALEUR DE LA FONCTION ECONOMIQUE et
2. RESTER LOIN DE LA FRONTIERE DU DOMAINE

LES “PATH FOLLOWING ALGORITHMS”

FAMILLE D'ALGORITHMES QUI ALLIENT DE BONNES PERFORMANCES THEORIQUES ET PRATIQUES. ELLES SONT FONDEES SUR 3 IDEES:

1. TRANSFORME LE PROGRAMME LINEAIRE EN UN PROBLEME NON CONTRAINT EN INCORPORANT LES CONTRAINTES DANS UNE FONCTION “BARRIERE LOGARITHMIQUE” QUI IMPOSE UNE PENALITE QUI VA EN CROISSANT AU FUR ET A MESURE QUE L’ON S’APPROCHE DE LA FRONTIERE DU DOMAINE (très similaire à ce qui est fait dans la méthode de réduction de potentiels).
2. ON RESOUT LE PROBLEME NON CONTRAINT DE BARRIERE LOGARITHMIQUE DE MANIERE APPROCHEE PAR L’ALGORITHME DE NEWTON
3. AU FUR ET A MESURE QUE LA FORCE DE LA FONCTION DE BARRIERE EST DIMINUEE, L’OPTIMUM DU PROBLEME NON CONTRAINT SUIV UN CERTAIN CHEMIN QUI ABOUTIT SUR UNE SOLUTION OPTIMALE

2 LA METHODE “AFFINE SCALING”

On omettra le symbole qui represente la transposition d’une matrice sauf si cela peut prêter à confusion et alors on note A^t la transposé de A . Les vecteurs sont donc consideres comme des matrices ligne ou/et colonne suivant les cas.

Soit le PL suivant:

$$\begin{array}{ll} \text{minimiser} & c.x \\ \text{sujet à:} & Ax = b \\ & x \geq 0 \end{array}$$

et son dual:

$$\begin{array}{ll} \text{maximiser} & p.b \\ \text{sujet à:} & pA \leq c \end{array}$$

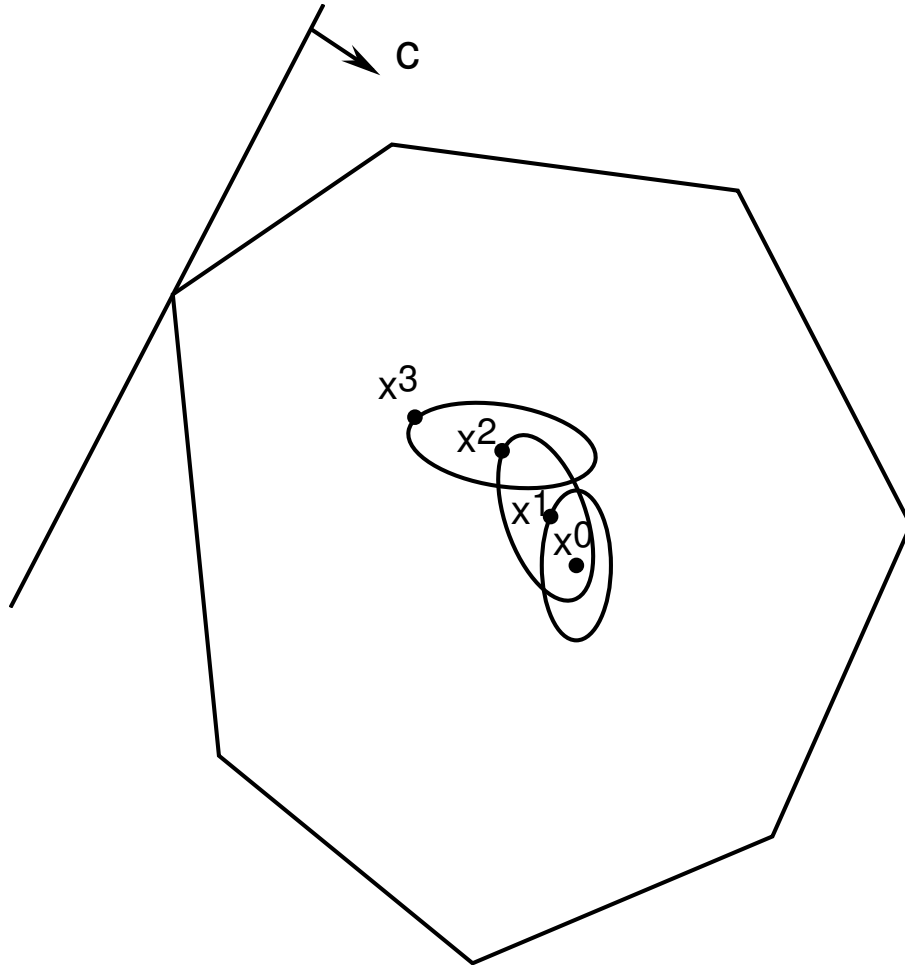
où A est une matrice $m \times n$.

$P = \{x : Ax = b, x \geq 0\}$ est l’ensemble des *solutions réalisables* et $\{x \in P : x > 0\}$ est l’*interieur* de P , ses éléments sont des *points intérieurs*

IDEE CENTRALE: minimiser cx pour $x \in P$ peut être difficile, par contre il est TRES FACILE de minimiser cx pour x appartenant à une ellipsoïde et de plus la solution possède une forme *analytique*. Donc au lieu de résoudre

directement sur P on va résoudre une suite de problèmes d'optimisation sur des ellipsoïdes.

- On commence en un point $x^0 > 0$ à l'intérieur de P . On forme une ellipsoïde S_0 centrée en x^0 et contenue dans l'intérieur de P . On optimise cx pour $x \in S_0$ ce qui donne un autre point intérieur x^1
- On construit une autre ellipsoïde centrée en x^1 ...



Ici x^1 minimise cx pour x dans l'ellipsoïde centrée en x^0 , x^2 minimise cx pour x dans l'ellipsoïde centrée en x^1

Lemme 1 Soit $\beta \in (0, 1)$ et $y \in \mathbb{R}^n$ tel que $y > 0$ et soit:

$$S = \{x \in \mathbb{R}^n : \sum_{i=1}^n \frac{(x_i - y_i)^2}{y_i^2} \leq \beta^2\}$$

Alors $x > 0$ pour tout $x \in S$.

Preuve. Soit $x \in S$. Pour tout i on a:

$(x_i - y_i)^2 \leq \beta^2 y_i^2 < y_i^2$ et donc $|x_i - y_i| < y_i$. Si $x_i - y_i \geq 0$ on a bien $x_i > 0$, sinon $-x_i + y_i < y_i$ et donc $x_i > 0$. \square

Fixons $y \in \mathbb{R}^n$ tel que $y > 0$ et $Ay = b$. Soit $Y = \text{diag}(y_1, \dots, y_n)$ la matrice diagonale dont les termes de la diagonale sont y_1, \dots, y_n . Y est inversible puisque les termes de la diagonales sont strictement positifs. Alors la relation $x \in S$ peut s'écrire

$\|Y^{-1}(x - y)\| \leq \beta$ où $\|\cdot\|$ représente la norme euclidienne. L'ensemble S est une ellipsoïde centrée en y . L'ensemble $S_0 = S \cap \{x : Ax = b\}$ est une section de l'ellipsoïde S et est lui même une ellipsoïde **contenue** dans le domaine des solutions réalisables.

On va minimiser sur S_0 :

$$\begin{aligned} \text{minimiser} \quad & cx \\ \text{sujet à:} \quad & Ax = b \\ & \|Y^{-1}(x - y)\| \leq \beta \end{aligned}$$

Posons $d = x - y$. On a $Ay = b$ et pour tout $x \in S_0$ on a aussi $Ax = b$, donc $Ad = 0$. Si on optimise sur d au lieu de x on a:

$$\begin{aligned} \text{minimiser} \quad & cd \\ \text{sujet à:} \quad & Ad = 0 \\ & \|Y^{-1}d\| \leq \beta \end{aligned} \tag{1}$$

Lemme 2 *Supposons que les lignes de A sont linéairement indépendantes et que c n'est pas une combinaison linéaire des lignes de A . Soit y un vecteur strictement positif, alors une solution optimale du problème (1) est donnée par:*

$$d^* = -\beta \frac{Y^2(c - A^t p)}{\|Y(c - A^t p)\|}$$

où

$$p = (AY^2 A^t)^{-1} AY^2 c$$

De plus le vecteur $x = y + d^* \in P$ et:

$$cx = cy - \beta \|Y(c - A^t p)\| < cy$$

Preuve. Pas donnée, mais à peine une page et sans difficulté majeure. \square

interprétation de la formule de p : Supposons que y soit une solution de base non dégénérée (ce qui n'est jamais le cas). Supposons que les m premières variables sont de base, alors $A = [B \ N]$.

Si $Y = \text{diag}(y_1, \dots, y_m, 0, \dots, 0)$ et $Y_0 = \text{diag}(y_1, \dots, y_m)$ alors $AY = [BY_0 \ 0]$ et

$$\begin{aligned} p &= (AY^2 A^t)^{-1} AY^2 c \\ &= (B^t)^{-1} Y_0^{-2} B^{-1} BY_0^2 c_B \\ &= (B^t)^{-1} c_B \end{aligned} \tag{2}$$

et donc p correspond à la solution duale de base correspondante. c'est pourquoi on appelle p les *estimations du dual*. De plus $r = c - A^t p$ devient $r = c - A^t (B^t)^{-1} c_B$ qui est le vecteur des coûts réduits du simplexe. Supposons r non négatif, alors p est dual réalisable et $r^t y = (c - A^t p)y = c^t y - p^t Ay = c^t y - p^t b$ la différence des valeurs des fonctions économiques du primal et du dual, appelé *saut de dualité*

Lemme 3 Soient y et p des solutions respectivement primale et duale réalisables telles que $cy - bp < \epsilon$

Soient y^* et p^* les solutions respectivement primale et duale optimales, alors:

$$\begin{aligned} cy^* &\leq cy < cy^* + \epsilon \\ bp^* - \epsilon &< bp \leq bp^* \end{aligned}$$

Preuve. Puisque y est réalisable on a $cy^* \leq cy$. Par dualité $bp \leq cy^*$. Puisque $cy - bp < \epsilon$, nous avons: $cy < bp + \epsilon \leq cy^* + \epsilon$.

De même on obtient: $bp^* = cy^* \leq cy < bp + \epsilon$ \square

Remarque 1 Si $d^* \geq 0$, le domaine des solutions du problème initial est non borné puisque $x + \alpha d^* > 0$ pour tout $\alpha > 0$ et $Ad^* = 0$. Puisque $cd < 0$ il en découle que le problème initial n'a pas de solution optimale finie.

Ceci suggère un critère d'arrêt. On s'arrête quand $r = c - A^t p \geq 0$ (solution duale réalisable) et $r^t y = y^t r = e^t Y r$ est petit, où $e = (1, 1, \dots, 1)$.

Le "Affine Scaling Algorithm" prend en entrée:

- (a) Les données: (A, b, c) ;
- (b) Une solution réalisable de départ $x^0 > 0$;
- (c) Une tolérance $\epsilon > 0$;
- (d) Le paramètre $\beta \in (0, 1)$.

1. (initialisation) On démarre avec une solution réalisable $x^0 > 0$, $k = 0$;

2. (Calcul des valeurs approchées du dual et des coûts réduits). Soit $x^k > 0$ réalisable:

$$\begin{aligned} X_k &= \text{diag}(x_1^k, \dots, x_n^k) \\ p^k &= (AX_k^2 A^t)^{-1} AX_k^2 c \\ r^k &= c - A^t p^k \end{aligned}$$

3. (test d'optimalité) Soit $e = (1, \dots, 1)$. Si $r^k \geq 0$ et $e^t X_k r^k < \epsilon$, STOP, x^k et p^k sont ϵ -optimales du primal et du dual resp.
4. (test de non "finitude") Si $-X_k^2 r^k \geq 0$, STOP solution optimale non finie.
5. (mise à jour de la solution primale) Poser:

$$x^{k+1} = x^k - \beta \frac{X_k^2 r^k}{\|X_k r^k\|} \quad (3)$$

2.1 VARIANTES DE L'ALGORITHME

Celles-ci diffèrent dans le choix de la longueur du pas. Etant donné un vecteur u on définit:

- $\|u\|_\infty = \max_i |u_i|$
- $\gamma(u) = \max\{u_i : u_i > 0\}$

Il est facile de vérifier que:

$$\gamma(u) \leq \|u\|_\infty \leq \|u\|$$

La version de l'algorithme que nous avons présentée est dite à *pas courts*. Dans les versions à *pas long* la mise à jour de la solution primale est remplacée par:

$$x^{k+1} = x^k - \beta \frac{X_k^2 r^k}{\|X_k r^k\|_\infty}$$

ou

$$x^{k+1} = x^k - \beta \frac{X_k^2 r^k}{\gamma(X_k r^k)}$$

Cette dernière est la plus utilisée car elle donne le pas le plus long et donc la décroissance la plus forte de la fonction économique. Dans tous les cas on peut montrer que $x^{k+1} > 0$ et réalisable.

A ce point on peut se poser plusieurs **QUESTIONS**

- (a) L'algorithme se termine-t-il?
- (b) Comment démarre-t-on?
- (c) Comment l'algorithme se comporte-t-il en pratique?

Il existe des théorèmes de convergence sur lesquels nous ne nous attarderons pas.

Pour **démarrer** on peut, par exemple, créer une nouvelle variable x_{n+1} et une nouvelle colonne $A_{n+1} = b - Ae$ où $e = (1, \dots, 1)$, et considérer le problème:

$$\begin{array}{ll} \text{minimiser} & cx + Mx_{n+1} \\ \text{sujet à:} & Ax + (b - Ae)x_{n+1} = b \\ & (x, x_{n+1}) \geq 0 \end{array}$$

où M est un grand nombre positif. Or $(e, 1)$ est une solution réalisable de ce problème, et si M est assez grand, dans la solution optimale on a $x_{n+1} = 0$ et donc cette solution optimale peut servir pour démarrer l'algorithme.

2.2 Performances pratiques

Le point critique est le calcul, à chaque itération de la matrice $AX_k^2A^t$ qui nécessite $O(m^2n)$ opérations, puis il faut résoudre un système linéaire, de matrice $AX_k^2A^t$, ce qui prend $O(m^3)$ opérations, donc chaque itération nécessite $O(m^2n + m^3)$ opérations et comme $m \leq n$ au total on a au plus $O(n^3)$ opérations par itération.

On a pu observer que si on démarre près de l'optimum, alors l'algorithme se comporte presque comme le simplexe et fait de petits pas car les ellipses sont petites. Par contre si on démarre loin, il fait de gros progrès vers l'optimum. C'est donc un bon candidat pour un algorithme mixte points intérieurs-simplexe (cf plus loin).

Exemple numérique 1:

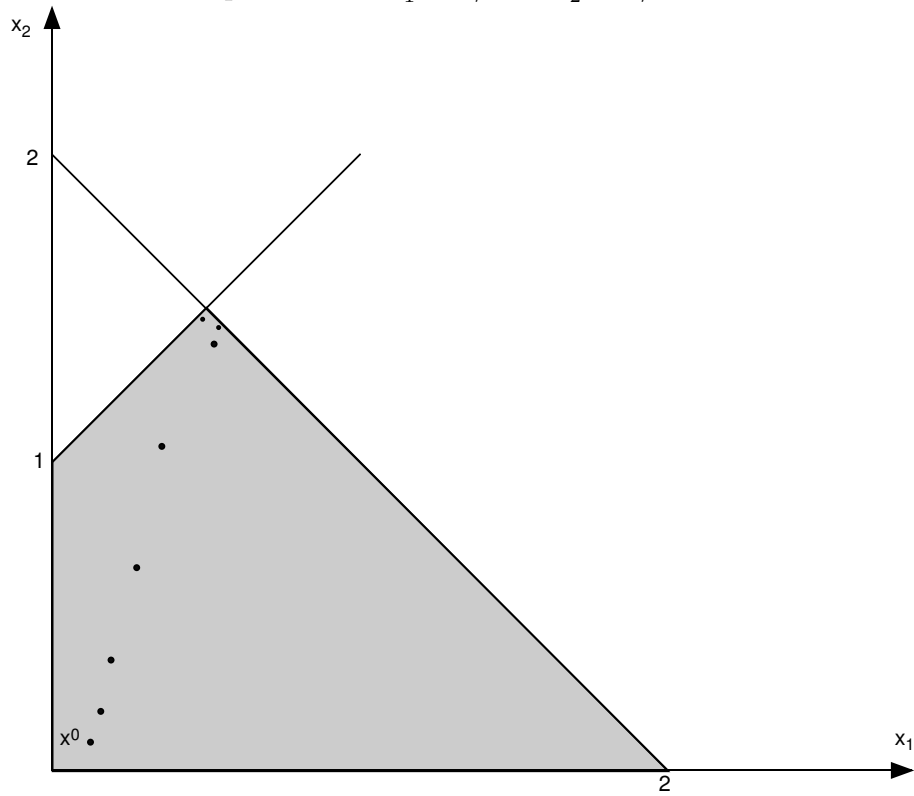
$$\begin{array}{llll} \text{maximiser} & x_1 + 2x_2 & & \\ \text{sujet à:} & & & (4) \\ x_1 + x_2 & \leq & 2 & \\ -x_1 + x_2 & \leq & 1 & \\ x_1, x_2 & \geq & 0 & \end{array}$$

Après addition de variables d'écart et transformation en un problème de minimisation:

$$\begin{array}{ll}
\text{minimiser} & -x_1 - 2x_2 \\
\text{sujet à:} & x_1 + x_2 + x_3 = 2 \\
& -x_1 + x_2 + x_4 = 1 \\
& x_1, x_2, x_3, x_4 \geq 0
\end{array}$$

x_1	.100	.144	.198	.262	0.364	0.530	0.546	0.499
x_2	.100	.188	.359	.667	1.068	1.339	1.439	1.491

Résultats d'itérations consécutives de la version "pas courts" avec $\beta = 0,995$. La solution optimale est $x_1^* = 1/2$ et $x_2^* = 3/2$.



3 LE “POTENTIAL REDUCTION ALGORITHM”

Soit le PL suivant:

$$\begin{array}{ll} \text{minimiser} & cx \\ \text{sujet à:} & Ax = b \\ & x \geq 0 \end{array}$$

et son dual:

$$\begin{array}{ll} \text{maximiser} & pb \\ \text{sujet à:} & pA + s = c \\ & s \geq 0 \end{array}$$

avec l’hypothèse suivante:

Hypothèse 1 *La matrice A a ses lignes linéairement indépendantes et il existe $x > 0$ et (p, s) avec $s > 0$, qui sont réalisables pour le primal et le dual respectivement.*

Le “affine scaling” algorithm décroissait strictement la fonction économique à chaque itération. Le résultat est que la séquence de solutions approche très rapidement de la frontière du domaine et à partir de là les progrès sont lents puisque les ellipsoïdes sont de plus en plus petites. Une solution est d’essayer de **repousser** les solutions de la frontière du domaine de façon à pouvoir continuer à faire des progrès significatifs.

On introduit la *fonction potentielle* $G(x, s)$

$$G(x, s) = q \log s \cdot x - \sum_{j=1}^n \log x_j - \sum_{j=1}^n \log s_j$$

où q est une constante plus grande que n . Si x et (p, s) sont réalisables du primal et du dual resp.

$$c \cdot x - b \cdot p = (s + pA) \cdot x - x^t A^t p = sx$$

Donc le premier terme mesure le saut de dualité, les deux autres termes pénalisent la proximité de la frontière du domaine pour la solution primale et duale respectivement.

Le théorème suivant montre que si on peut décroître, à chaque itération, la valeur de la fonction potentielle $G(x, s)$ d’une certaine quantité, on peut garantir d’obtenir une solution ϵ -optimale après un petit nombre d’itérations.

Théorème 4 Soient $x^0 > 0$ et (p^0, s^0) avec $s^0 > 0$, des solutions resp. primale et duale réalisables. Soit $\epsilon > 0$ le seuil de tolérance pour l'optimum. Tout algorithme qui conserve la réalisabilité primale et duale et qui fait décroître $G(x, s)$ d'une quantité supérieure ou égale à $\delta > 0$ à chaque itération, trouve une solution des problèmes primal et du dual de saut dual:

$$s^K . x^K \leq \epsilon$$

après au plus

$$K = \left\lceil \frac{G(x^0, s^0) + (q - n) \log(1/\epsilon) - n \log n}{\delta} \right\rceil$$

itérations

Preuve. On a:

$$\begin{aligned} G(x, s) &= q \log sx - \sum_{j=1}^n \log x_j - \sum_{j=1}^n \log s_j \\ &= n \log sx - \sum_{j=1}^n \log x_j - \sum_{j=1}^n \log s_j + (q - n) \log sx \\ &\geq n \log n + (q - n) \log sx \end{aligned} \tag{5}$$

Car $n \log sx - \sum_{j=1}^n \log x_j - \sum_{j=1}^n \log s_j$ atteint son minimum quand $x_j s_j = sx/n$, ce qui peut être vu en annulant la dérivée et en vérifiant que la dérivée seconde est non négative. Donc,

$$n \log sx - \sum_{j=1}^n \log x_j - \sum_{j=1}^n \log s_j \geq n \log n \tag{6}$$

Soit $\delta > 0$ fixé et supposons que nous avons un algorithme qui a la propriété que:

$$G(x^{k+1}, s^{k+1}) - G(x^k, s^k) \leq -\delta, \forall k$$

Après K itérations on a:

$$G(x^K, s^K) - G(x^0, s^0) \leq -K\delta.$$

Pour la valeur de K du théorème on a

$$G(x^K, s^K) \leq -(q - n) \log \frac{1}{\epsilon} + n \log n$$

En utilisant l'inégalité (5), on a le saut de dualité en dessous du seuil désiré.

$$s^K . x^K \leq \epsilon$$

□

Ce théorème nous incite à construire un algorithme qui décroît la fonction potentielle $G(x, s)$ d'au moins une constante à chaque itération. C'est pour cette raison que l'algorithme que nous allons décrire s'appelle "de réduction de potentiel".

Intuitivement, l'idée centrale est la suivante: En partant d'une solution réalisable $x > 0$ du primal et d'une solution réalisable du dual avec $s > 0$, on cherche une direction d telle que $G(x + d, s) < G(x, s)$. La direction d doit satisfaire:

$$Ad = 0, \quad \|X^{-1}d\| \leq \beta < 1$$

de façon à ce que $x + d$ soit réalisable, comme vu dans la méthode "affine scaling". Le problème de minimiser $G(x + d, s)$ sous les contraintes précédentes est un problème non linéaire très difficile. Pour cette raison, on approxime la fonction potentiel $G(x + d, s)$ par son développement de Taylor au premier ordre et on résout le problème suivant:

$$\begin{array}{ll} \text{minimiser} & \nabla_x G(x, s)^t d \\ \text{sujet à} & Ad = 0 \\ & \|X^{-1}d\| \leq \beta \end{array}$$

Le problème est exactement le même que celui rencontré dans la méthode précédente, sauf pour la fonction économique dont les coeffs sont $\hat{c} = \nabla_x G(x, s)$ au lieu de c , i.e.

$$\hat{c}_i = \frac{\partial G(x, s)}{\partial x_i} = \frac{qs_i}{sx} - \frac{1}{x_i}$$

Donc en appliquant le Lemme 2 on obtient:

$$d^* = -\beta X \frac{u}{\|u\|}$$

où:

$$u = X(\hat{c} - A^t(AX^2A^t)^{-1}AX^2\hat{c})$$

Puisque:

$$X\hat{c} = \frac{q}{sx}Xs - e$$

on obtient:

$$u = (I - XA^t(AX^2A^t)^{-1}AX) \left(\frac{q}{sx}Xs - e \right)$$

De plus, $G(x, s)$ décroît de $\beta \|u\| + O(\beta^2)$, où le premier terme vient du Lemme 2 et le deuxième est dû aux termes d'ordre supérieurs omis dans le développement de Taylor de $G(x, s)$.

En bornant soigneusement les termes d'ordre supérieurs, on peut montrer que si $\|u\| \geq \gamma$, pour un γ donné, alors la fonction potentiel diminue d'au moins une valeur constante.

Remarquez que dans une telle itération ni s ni p ne sont modifiés. Cependant, si $\|u\| < \gamma$, on ne peut pas décroître suffisamment la fonction potentiel, dans ce cas on change les variables duales pour obtenir la décroissance voulue.

L'algorithme prend en entrée:

- (a) Les données: (A, b, c) ; la matrice est de rang m .
- (b) Des solution réalisables de départ $x^0 > 0, s^0 > 0, p^0$;
- (c) Une tolérance d'optimalité $\epsilon > 0$;
- (d) Les paramètres $\beta \in (0, 1)$, γ et q .

1. (initialisation) On démarre avec une solution réalisable $x^0 > 0, s^0 > 0, p$. Poser $k = 0$;
2. (test d'optimalité) Si $s^k \cdot x^k < \epsilon$ STOP
3. (Calcul de la direction).

$$\begin{aligned}
X_k &= \text{diag}(x_1^k, \dots, x_n^k) \\
\bar{A}^k &= (AX_k)^t (AX_k^2 A^t)^{-1} AX_k \\
u^k &= (I - \bar{A}^k) \left(\frac{q}{s^k \cdot x^k} X_k s^k - e \right) \\
d^k &= -\beta X_k u^k / \|u^k\|
\end{aligned}$$

4. (itération primale) Si $\|u^k\| \geq \gamma$, poser:

$$\begin{aligned}
x^{k+1} &= x^k + d^k \\
s^{k+1} &= s^k \\
p^{k+1} &= p^k
\end{aligned}$$

5. (itération duale) Si $\|u^k\| < \gamma$, poser:

$$\begin{aligned}
x^{k+1} &= x^k \\
s^{k+1} &= \frac{s^k \cdot x^k}{q} X_k^{-1} (u^k + e) \\
p^{k+1} &= p^k + (AX_k^2 A^t)^{-1} AX_k \left(X_k s^k - \frac{s^k \cdot x^k}{q} e \right)
\end{aligned}$$

6. Poser $k=k+1$ et retourner en (2).

On peut montrer que x^k et (p^k, s^k) sont toujours primale et duale réalisables. Il n'y a malheureusement pas d'interprétation intuitive de l'itération duale. Nous terminerons par le théorème suivant qui montre que la décroissance voulue peut être atteinte:

Théorème 5 *L'algorithme avec $\beta < 1$ et $\gamma < 1$ a les propriétés suivantes:*

- (a) Si $\|u^k\| \geq \gamma$ (itération primale), alors

$$G(x^{k+1}, x^{s+1}) - G(x^k, x^s) \leq -\beta\gamma + \frac{\beta^2}{2(1-\beta)}$$

(b) Si $\|u^k\| < \gamma$ (itération duale), alors:

$$G(x^{k+1}, x^{s+1}) - G(x^k, x^s) \leq -(q - n) + n \frac{q}{n} + \frac{\gamma^2}{2(1-\gamma)}$$

(c) Si $q = n + \sqrt{n}$, $\beta \approx 0,285$ et $\gamma \approx 0,479$, alors l'algorithme décroît $G(x, s)$ d'au moins $\delta = 0,079$ à chaque itération.

Preuve. Démonstration longue et fastidieuse. \square

Initialisation

On montre dans cette section comment initialiser. On considère le programme artificiel:

$$\begin{aligned} \text{minimiser} \quad & cx + M_1 x_{n+1} \\ \text{sujet à:} \quad & Ax + (b - Ae)x_{n+1} = b \\ & (e - c).x + x_{n+2} = M_2 \\ & x_1, \dots, x_{n+2} \geq 0 \end{aligned}$$

et son dual:

$$\begin{aligned} \text{maximiser} \quad & pb + p_{m+1}M_2 \\ \text{sujet à:} \quad & pA + p_{m+1}(e - c) + s = c \\ & p.(b - Ae) + s_{n+1} = M_1 \\ & p_{m+1} + s_{n+2} = 0 \\ & s_1, \dots, s_{n+2} \geq 0 \end{aligned}$$

Les variables x_{n+1} , x_{n+2} , p_{m+1} , s_{n+1} et s_{n+2} sont des variables artificielles, M_1 et M_2 sont grands et spécifiés plus loin. On doit avoir:

$$M_2 > (e - c).e.$$

Les vecteurs:

$$\begin{aligned} (x^0, x_{n+1}^0, x_{n+2}^0) &= (e, 1, M_2 - (e - c).e) \\ (p^0, p_{m+1}^0, s^0, s_{n+1}^0, s_{n+2}^0) &= (\mathbf{0}, -1, e, M_1, 1) \end{aligned}$$

sont réalisables pour les problèmes artificiaux primal et dual resp. et peuvent être utilisés pour démarrer l'algorithme de réduction de potentiel. La relation entre le problème initial et le problème artificiel est donnée par:

Theorème 6 Soient x^* et (p^*, s^*) des solutions optimales du primal et du dual resp., dont l'existence qui est supposée. Si:

$$M_1 \geq \max\{(b - Ae)^t p^*, 0\} + 1$$

et:

$$M_2 \geq \max\{(e - c)^t x^*, (e - c)^t e, 0\} + 1$$

alors:

1. Une solution réalisable $(\bar{x}, \bar{x}_{n+1}, \bar{x}_{n+2})$ du problème artificiel primal est optimale **ssi** \bar{x} est optimale du problème initial et $\bar{x}_{n+1} = 0$
2. Une solution réalisable $(\bar{p}, \bar{p}_{m+1}, \bar{s}, \bar{s}_{n+1}, \bar{s}_{n+2})$ du problème artificiel dual est une solution optimale **ssi** (\bar{p}, \bar{s}) est optimale du problème dual initial et $\bar{p}_{m+1} = 0$

Preuve. (1) Soit $(\bar{x}, \bar{x}_{n+1}, \bar{x}_{n+2})$ une solution optimale du problème artificiel primal. Supposons que $\bar{x}_{n+1} > 0$. A partir de x^* définissons la solution réalisable suivante du problème artificiel primal, en posant $x_{n+1}^* = 0$ et $x_{n+2}^* = M_2 - (e - c).x^*$. Alors

$$c.x^* + M_1 x_{n+1}^* = p^*.b = p^*. (A\bar{x} + (b - Ae)\bar{x}_{n+1})$$

Puisque $(p^*)^t A + (s^*)^t = c^t$, $\bar{x}_{n+1} > 0$ et $M_1 > (b - Ae)^t p^*$ on a:

$$c.x^* + M_1 x_{n+1}^* < (c - s)^t \bar{x} + M_1 \bar{x}_{n+1} \leq c.\bar{x} + M_1 \bar{x}_{n+1}$$

parce que $s^*.\bar{x} \geq 0$, ce qui contredit l'optimalité de $(\bar{x}, \bar{x}_{n+1}, \bar{x}_{n+2})$. De plus, l'inégalité précédente montre que $(x^*, x_{n+1}^*, x_{n+2}^*)$ est optimale pour le problème artificiel primal et que sa valeur est $c.\bar{x} = c.x^*$. Puisque \bar{x} est réalisable, elle est optimale pour le problème primal original.

Réciproquement, soit x^* une solution optimale du primal original. Alors $(\bar{x}, \bar{x}_{n+1}, \bar{x}_{n+2})$ avec $\bar{x} = x^*$, $\bar{x}_{n+1} = 0$, $\bar{x}_{n+2} = M_2 - (e - c).x^*$ est réalisable pour le primal artificiel. Sa valeur $c.\bar{x} + M_1 \bar{x}_{n+1}$ coïncide avec sa valeur optimale $c.x^* + M_1 x_{n+1}^*$ et donc est optimale du primal artificiel. \square

3.1 Complexité de l'algorithme

On suppose que A , b et c ont des composantes entières dont les valeurs absolues sont bornées par U . Nous n'entrerons pas dans les détails. Il est clair que la valeur de la solution initiale compte. On peut montrer que $G(x^0, s^0) = O(qn \log(nU))$. D'après les théorèmes 4 et 5 avec $q = n + \sqrt{n}$ et la valeur $G(x^0, s^0)$ ci-dessus, on peut conclure que l'algorithme trouve des solutions x^K et s^K avec un saut de dualité:

$$(s^K).x^K \leq \epsilon \tag{7}$$

après:

$$K = 0 \left(\sqrt{n} \log \frac{1}{\epsilon} + n^2 \log(nU) \right) \tag{8}$$

itérations.

Chaque itération nécessite $O(nm^2 + m^3)$ opérations arithmétiques. Comme $m \leq n$, on a au plus $O(n^3)$ opérations arithmétiques. Donc l'algorithme trouve une solution ϵ -optimale en :

$$O\left(n^{3.5} \log \frac{1}{\epsilon} + n^5 \log(nU)\right) \quad (9)$$

opérations arithmétiques. Notez que cet algorithme est **POLYNOMIAL**. Il y a des améliorations pour accélérer la convergence.

4 LE “PRIMAL PATH FOLLOWING ALGORITHM”

C'est la méthode qui semble la plus employée et la plus expérimentée. Elle donne de très bons résultats tant théoriques que pratiques.

Cet algorithme résout le PL suivant :

$$\begin{array}{ll} \text{minimiser} & c.x \\ \text{sujet à:} & Ax = b \\ & x \geq 0 \end{array}$$

et son dual :

$$\begin{array}{ll} \text{maximiser} & p.b \\ \text{sujet à:} & pA + s = c \\ & s \geq 0 \end{array}$$

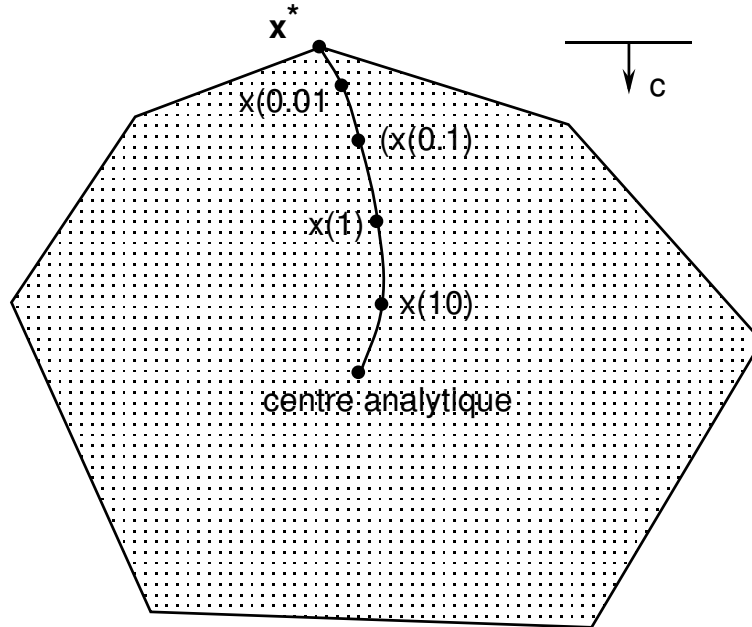
Une des idées derrière cette méthode est l'observation qu'une des difficultés de la programmation linéaire provient des inéquations $x \geq 0$. En effet les variables non astreintes, une fois dans la base, n'en sortent jamais et de plus dès qu'un coût réduit est non nul dans la fonction économique, elle peut entrer dans la base. Pour cette raison on va convertir le PL en un problème avec seulement des équations, en utilisant une fonction *barrière* qui va empêcher une variable d'atteindre la frontière $x_j = 0$. On y parvient en ajoutant les termes $-\log x_j$ à la fonction économique. Ces termes vont faire augmenter cette fonction vers $+\infty$ quand x_j va tendre vers 0. Nous introduisons la *fonction barrière* suivante :

$$B_\mu(x) = c.x - \mu \sum_{i=1}^n \log x_j$$

supposée valoir l'infini si $x_j \leq 0$ pour au moins un j . On considère la famille de programmes **non** linéaires (dits *problèmes barrières*) suivante:

$$\begin{aligned} &\text{minimiser} && B_\mu(x) \\ &\text{sujet à:} && Ax = b \end{aligned} \tag{10}$$

Supposons que pour chaque $\mu > 0$, le problème barrière possède une solution optimale $x(\mu)$. (On peut facilement montrer que ce problème ne peut pas avoir de solutions optimales multiples car le domaine est strictement convexe.) Au fur et à mesure que μ varie, les solutions $x(\mu)$ successives forment le “*chemin central*” (“*central path*”). Celui-ci est illustré sur la figure suivante.



On peut montrer que $\lim_{\mu \rightarrow 0} x(\mu)$ existe et est une solution optimale x^* du programme linéaire initial. L'idée intuitive est que quand μ est très petit, le terme logarithmique est négligeable presque partout, mais il nous empêche quand même d'atterrir sur la frontière.

Un problème barrière ayant comme origine le dual est le suivant:

$$\begin{aligned} &\text{maximiser} && p.b + \mu \sum_{j=1}^n \log s_j \\ &\text{sujet à:} && pA + s = c \end{aligned} \tag{11}$$

Soient $p(\mu)$, $s(\mu)$ une solution optimale de ce problème (11) pour $\mu > 0$. Les problèmes (10) et (11) sont des problèmes d'optimisation convexe, c.à d.

que l'on optimise une fonction convexe sous des contraintes qui définissent un ensemble convexe (ici des inéquations linéaires). Il existe des conditions similaires aux écarts complémentaires de la programmation linéaire, ce sont les conditions de Karush-Kuhn-Tucker. Elles sont nécessaires et suffisantes pour que des solutions soient optimales pour les problèmes (10) et (11)

Conditions de KKT:

$$\begin{aligned} Ax(\mu) &= b \\ x(\mu) &\geq 0 \\ A^t p(\mu) + s(\mu) &= c \\ s(\mu) &\geq 0 \\ X(\mu)S(\mu)e &= e\mu \end{aligned} \tag{12}$$

où $X(\mu) = \text{diag}(x_1(\mu), \dots, x_n(\mu))$ et $S(\mu) = \text{diag}(s_1(\mu), \dots, s_n(\mu))$. Noter que si $\mu = 0$, alors ce sont les conditions du théorème des écarts complémentaires.

Lemme 7 Si x^* , p^* et s^* satisfont les condition (12), alors ils sont solutions optimales des problèmes (10) et (11), c.à d.

$$x^* = x(\mu), \quad p^* = p(\mu), \quad s^* = s(\mu)$$

Preuve. Démonstration très facile. \square

Si $\mu = \infty$, le problème devient:

$$\begin{aligned} \text{minimiser} \quad & -\sum_{j=1}^n \log x_j \\ \text{sujet à:} \quad & Ax = b \end{aligned}$$

et sa solution optimale s'appelle le *centre analytique* du domaine des solutions.

Exemple numérique 2: Considérons le problème:

$$\begin{aligned} \text{minimiser} \quad & x \\ \text{sujet à:} \quad & x \geq 0 \end{aligned}$$

La fonction barrière est dans ce cas:

$$B_\mu x = x - \mu \log x \tag{13}$$

et la solution optimale est $x(\mu) = \mu$, et quand $\mu \rightarrow 0$, la solution tend vers la solution optimale $x^* = 0$

Exemple numérique 3: (Calcul du chemin central)

$$\begin{aligned} \text{minimiser} \quad & x_2 \\ \text{sujet à:} \quad & x_1 + x_2 + x_3 = 1 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

Pour calculer le chemin central on doit résoudre:

$$\begin{aligned} \text{minimiser} \quad & x_2 - \mu \log x_1 - \mu \log x_2 - \mu \log x_3 \\ \text{sujet à:} \quad & x_1 + x_2 + x_3 = 1 \end{aligned}$$

En substituant $x_3 = 1 - x_1 - x_2$, on doit résoudre le problème non linéaire non contraint suivant:

$$\min x_2 - \mu \log x_1 - \mu \log x_2 - \mu \log(1 - x_1 - x_2)$$

En annulant les dérivées on trouve:

$$\begin{aligned} x_1(\mu) &= \frac{1 - x_2(\mu)}{2} \\ x_2(\mu) &= \frac{1 + 3\mu - \sqrt{1 + 9\mu^2 + 2\mu}}{2} \\ x_3(\mu) &= \frac{1 - x_2(\mu)}{2} \end{aligned}$$

Le centre analytique est trouvé en faisant $x \rightarrow \infty$. C'est le point $(1/3, 1/3, 1/3)$.

L'ensemble des solutions optimales du programme linéaire est:

$$Q = \{x : x = (x_1, 0, x_3), x_1 + x_3 = 1, x \geq 0\}$$

dont le centre analytique est $(1/2, 0, 1/2)$, qui est le point obtenu si on pose $\mu = 0$. Le chemin central aboutit sur le centre analytique des solutions optimales.

Remarque 2 *Le centre analytique n'est pas une notion géométrique, elle dépend des inéquations utilisées pour décrire le domaine des solutions. Le rajout d'une contrainte redondante change ce centre.*

Le problème barrière est très difficile à résoudre parce que sa fonction économique n'est ni linéaire ni quadratique. Pour cela on approxime $B_\mu(x)$ autour d'un point donné x par les trois premiers termes de son développement de Taylor. On a:

$$\begin{aligned} \frac{\partial B_\mu(x)}{\partial x_i} &= c_i - \frac{\mu}{x_i} \\ \frac{\partial^2 B_\mu(x)}{\partial x_i^2} &= \frac{\mu}{x_i^2} \\ \frac{\partial^2 B_\mu(x)}{\partial x_i \partial x_j} &= 0, \quad i \neq j \end{aligned}$$

Etant donné un vecteur $x > 0$, le développement de Taylor de la fonction barrière est:

$$\begin{aligned} B_\mu(x+d) &\approx B_\mu(x) + \sum_{i=1}^n \frac{\partial B_\mu(x)}{\partial x_i} d_i + \frac{1}{2} \sum_{i,j=1}^n \frac{\partial^2 B_\mu(x)}{\partial x_i \partial x_j} d_i d_j \\ &= B_\mu(x) + (c - \mu e X^{-1})d + \frac{1}{2} \mu d^t X^{-2} d \end{aligned}$$

Alors au lieu de minimiser la fonction barrière, on va chercher une direction d qui minimise le développement de Taylor de $B_\mu(x+d)$. Le problème d'approximation devient:

$$\begin{aligned} \text{minimiser} \quad & (c - \mu e X^{-1})d + \frac{1}{2} \mu d^t X^{-2} d \\ \text{sujet à:} \quad & Ad = 0 \end{aligned}$$

Ce problème peut être résolu de façon analytique en utilisant les multiplicateurs lagrangiens. On associe un vecteur p de multiplicateurs de lagrange aux contraintes $Ad = 0$ et on forme la fonction de Lagrange:

$$L(d, p) = (c^t - \mu e^t X^{-1})d + \frac{1}{2} \mu d^t X^{-2} d - p^t Ad$$

et on veut

$$\frac{\partial L(d, p)}{\partial d_j} = 0, \quad \frac{\partial L(d, p)}{\partial p_i} = 0$$

ce qui donne:

$$\begin{aligned} c - \mu X^{-1}e + \mu X^{-2}d - A^t p &= 0 \\ Ad &= 0 \end{aligned}$$

C'est un système linéaire de $m+n$ équations à $m+n$ inconnues (les d_j et les p_j) dont la solution est:

$$\begin{aligned} d(\mu) &= (I - X^2 A^t (A X^2 A^t)^{-1} A) (X e - \frac{1}{\mu} X^2 c) \\ p(\mu) &= (A X^2 A^t)^{-1} A (X^2 c - \mu X e) \end{aligned}$$

Le vecteur $d(\mu)$ s'appelle la *direction de Newton* et son calcul s'appelle une *itération de Newton*.

La solution x devient $x + d(\mu)$ et la solution duale (p, s) devient $(p(\mu), c - A^t p(\mu))$. Puis on décroît μ à $\bar{\mu} = \alpha \mu$, où $0 < \alpha < 1$ est fixé une fois pour toute.

En résumé la méthode est: 1) (Initialisation) Partir de solutions primale et duale réalisables $x^0 > 0$, $s^0 > 0$ et p^0 . $k = 0$.

2) (test d'optimalité) Si $s^k \cdot x^k < \epsilon$ STOP

3) Soit:

$$\begin{aligned} X_k &= \text{diag}(x_1^k, \dots, x_n^k) \\ \mu^{k+1} &= \alpha \mu^k \end{aligned}$$

4) (calcul des directions) Résoudre le système linéaire:

$$\begin{aligned} \mu^{k+1} X_k^{-2} d - A^t p &= \mu^{k+1} X_k^{-1} e - c \\ Ad &= 0 \end{aligned}$$

pour trouver p et d

5) (mise à jour des solutions) Poser:

$$\begin{aligned} x^{k+1} &= x^k + d \\ p^{k+1} &= p \\ s^{k+1} &= c - A^t p \end{aligned}$$

$k=k+1$, aller en 2.