

Chapitre 4

Méthodes de descente. Problèmes sans contraintes

Sommaire

4.1	Principe	35
4.2	Méthode de relaxation	36
4.3	Méthode du gradient	36
4.3.1	Méthode à pas variable	36
4.3.2	Méthode à pas optimal	36
4.4	Estimations et convergence dans le cas quadratique	37
4.4.1	Méthode à pas optimal	37
4.4.2	Méthode de gradient à pas constant	38
4.5	Méthode du gradient conjugué	38
4.5.1	Principe de la méthode	38
4.5.2	Ecriture comme algorithme de descente	38
4.5.3	Analyse de convergence	39
4.6	Calcul du pas pour les méthodes de descente	40
4.7	Méthodes de Newton et quasi-Newton	40

4.1 Principe

On se place dans un espace de Hilbert V , et on cherche à calculer numériquement un x (qui n'est pas forcément unique) tel que

$$\forall y \in V, J(x) \leq J(y) \tag{4.1}$$

Le principe est de construire un algorithme itératif de la forme

$$x^{k+1} = x^k - \rho_k d^k \tag{4.2}$$

d^k est la **direction de descente**, ρ_k est le **pas**. Il est, soit fixé, éventuellement le même pour toutes les étapes (on parle alors de **méthode à pas variable**), soit calculé à chaque étape de façon à minimiser J dans la direction d^k (on parle alors de **méthode à pas optimal**).

4.2 Méthode de relaxation

On se place en dimension finie, i.e. $V = \mathbb{R}^n$. Pour passer de x^k à x^{k+1} , on minimise successivement dans les n directions de la base canonique.

1. $x^{k,1}$ est défini par

$$J(x^{k,1}) = \inf_{\rho \in \mathbb{R}} J(x^k - \rho e_1)$$

ou encore

$$x^{k,1} = (x_1^k - \rho_1, x_2^k, \dots, x_n^k)$$

On note $x_1^{k+1} = x_1^k - \rho_1$

2. à l'étape i on a

$$x^{k,i} = (x_1^{k+1}, \dots, x_i^{k+1}, x_i^k, \dots, x_n^k)$$

$x^{k,i+1}$ est maintenant défini par

$$J(x^{k,i+1}) = \inf_{\rho} J(x^{k,i} - \rho e_{i+1})$$

3. $x^{k+1} = x^{k,n}$

Théorème 4.1 . Si J est α -convexe différentiable sur \mathbb{R}^n , si J' est uniformément lipschitzienne sur les bornés, l'algorithme de relaxation est bien défini et converge vers la solution optimale.

Remarque 4.1 . Dans le cas où J est quadratique, i.e. $J(v) = \frac{1}{2}(Av, v) - (b, v)$, on retrouve l'algorithme de Gauss-Seidel ou S.O.R. pour la résolution du système linéaire $Ax = b$.

4.3 Méthode du gradient

Ici on choisit à chaque étape $d^k = \nabla J(x^k)$.

4.3.1 Méthode à pas variable

On se donne le pas ρ_k , il peut être différent d'une étape à l'autre.

Théorème 4.2 . Supposons J α -convexe dérivable sur V , et ∇J uniformément lipschitzien de constante de Lipschitz M . Si il existe deux constantes a et b telles que pour tout $k > 0$, $0 < a \leq \rho_k \leq b < \frac{2\alpha}{M^2}$, l'algorithme de gradient à pas variable converge vers la solution optimale.

Remarque 4.2 . Si J est 2 fois différentiable, l'hypothèse est

$$\sup_{v \in V} \|D^2 J(v)\| \leq M$$

4.3.2 Méthode à pas optimal

Ici on choisit à chaque étape ρ_k de façon que

$$J(x^k - \rho_k \nabla J(x^k)) = \inf_{\rho \in \mathbb{R}} J(x^k - \rho \nabla J(x^k)) \quad (4.3)$$

Théorème 4.3 . Si J est α -convexe dérivable sur V , si ∇J est uniformément lipschitzien de constante de Lipschitz M , l'algorithme de gradient à pas optimal est bien défini et converge vers la solution optimale.

Remarque 4.3 . Les directions de descente sont orthogonales, i.e.

$$\nabla J(x^k) \cdot \nabla J(x^{k+1}) = 0.$$

4.4 Estimations et convergence dans le cas quadratique

Ici la fonctionnelle J est quadratique sur \mathbb{R}^n :

$$J(v) = \frac{1}{2}(Av, v) - (b, v)$$

où la matrice A est symétrique définie positive. La solution x du problème de minimisation vérifie $Ax = b$. On appellera **résidu** à l'étape k la quantité $r^k = b - Ax^k$.

4.4.1 Méthode à pas optimal

On prend ici une direction de descente d^k quelconque dans \mathbb{R}^n , non orthogonale à r^k . A chaque étape, la valeur du paramètre optimal ρ_k est donnée par

$$\rho_k = -\frac{(r^k, d^k)}{(Ad^k, d^k)} \quad (4.4)$$

et l'on a $(r^{k+1}, d^k) = 0$.

Notons $E(v) = \frac{1}{2}(A(v - u), v - u)$, on a alors

$$E(x^{k+1}) = (1 - \gamma_k)E(x^k) \quad (4.5)$$

avec

$$\gamma_k = \frac{(r^k, d^k)^2}{(Ad^k, d^k)(A^{-1}r^k, r^k)}. \quad (4.6)$$

Par construction $0 \leq \gamma_k \leq 1$. De plus, notant λ_j les valeurs propres de A , ordonnées, $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$, on a l'estimation suivante :

$$\gamma_k \geq \frac{\lambda_1}{\lambda_n} \left(\frac{r^k}{\|r^k\|}, \frac{r^k}{\|r^k\|} \right)^2$$

si la direction de descente est telle que

$$\left(\frac{r^k}{\|r^k\|}, \frac{d^k}{\|d^k\|} \right)^2 \geq \mu > 0 \quad (4.7)$$

alors $\gamma_k \geq \gamma = \frac{\mu}{K(A)}$ (où $K(A)$ est le conditionnement de A pour la norme euclidienne, c'est-à-dire le rapport de la plus grande à la plus petite valeur propre), et donc

$$E(x^{k+1}) \leq (1 - \gamma)E(x^k) \quad (4.8)$$

On dit que la méthode **converge linéairement**.

Dans le cas particulier de la méthode du gradient, $d^k = \nabla J(x^k)$, grâce à l'**inégalité de Kantorovitch** on peut écrire

$$\tau = \frac{K(A) - 1}{K(A) + 1}, \quad E(x) = \frac{1}{2}(A(x - E(x^{k+1})) \leq \tau^2 E(x^k), \quad E(x^k) \leq \left(\frac{K(A) - 1}{K(A) + 1} \right)^{2k} E(x^0) \quad (4.9)$$

Remarque 4.4 . Plus la matrice est bien conditionnée (i.e. $K(A)$ proche de 1), plus la convergence est rapide. Plus la matrice est mal conditionnée (i.e. $K(A) \gg 1$), plus la convergence est lente.

Lemme 4.1 (Lemme de Kantorovitch) *Pour toute matrice A symétrique définie positive, on a l'inégalité :*

$$\forall y \in \mathbb{R}^m \setminus 0, \quad \frac{(Ay, y)(A^{-1}y, y)}{(y, y)^2} \leq \frac{(\lambda_1 + \lambda_n)^2}{4\lambda_1\lambda_n} = \frac{((K(A)^{\frac{1}{2}} + K(A)^{-\frac{1}{2}})^2)}{4}.$$

4.4.2 Méthode de gradient à pas constant

On choisit à chaque étape $\rho_k = \rho$. On a alors l'estimation

$$\|x^k - x\|_2 \leq \left[\max_{1 \leq i \leq n} |1 - \rho\lambda_i| \right]^k \|x^0 - x\|_2 \quad (4.10)$$

On en déduit que la méthode converge si et seulement si $\rho < \frac{2}{\lambda_n}$ où λ_n est la plus grande valeur propre de A . Ici encore, la convergence est linéaire.

Le meilleur ρ est égal à $\frac{2}{\lambda_1 + \lambda_n}$.

4.5 Méthode du gradient conjugué

On se place ici dans le cas où la fonctionnelle J est quadratique sur \mathbb{R}^N : $J(v) = \frac{1}{2}(Av, v) - (b, v)$, la matrice A étant symétrique définie positive. La solution x du problème de minimisation vérifie $Ax = b$.

4.5.1 Principe de la méthode

Les $(k + 1)$ premières itérées x^0, \dots, x^k étant données, on cherche x^{k+1} , non plus dans la direction du gradient, mais dans l'espace vectoriel engendré par tous les gradients précédents. On note

$$\mathcal{L}_k = \text{vect}\{\nabla J(x^0), \dots, \nabla J(x^k)\} \quad (4.11)$$

et on définit x^{k+1} par :

$$J(x^{k+1}) = \inf_{\Delta \in \mathcal{L}_k} J(x^k + \Delta) \quad (4.12)$$

Ceci définit $x^{k+1} = x^k + \Delta^k$ de manière unique (cf Corollaire 1.1, Partie I) et

Théorème 4.4 . *On a les propriétés suivantes :*

1. Les $\nabla J(x^k)$ forment un système orthogonal (donc libre),
2. l'algorithme converge en au plus N itérations.

La première propriété traduit l'équation d'Euler (2.3, Partie I). Ce théorème nous dit que la méthode du gradient conjugué est en fait une méthode directe. La forme (4.12) n'est pas pratique, aussi allons nous réécrire l'algorithme sous forme d'un algorithme de descente.

4.5.2 Ecriture comme algorithme de descente

Lemme 4.2 *On a les propriétés suivantes*

1. Pour tout $\ell \neq 0$, $\Delta_\ell \neq 0$,
2. Les directions Δ_ℓ sont conjuguées par rapport à A , i.e.

$$\forall \ell, m, \quad \ell \neq m, \quad (A\Delta_\ell, \Delta_m) = 0.$$

On développe maintenant les directions Δ_ℓ dans la base des résidus

$$\begin{aligned}\Delta^0 &= \delta_0^0 r^0, \\ &\vdots \\ \Delta^k &= \delta_k^k r^k + \dots + \delta_k^0 r^0, \\ &\vdots\end{aligned}$$

et on calcule les coefficients

Théorème 4.5 . *L'algorithme du gradient conjugué s'écrit sous la forme*

$$\begin{cases} x^{k+1} = x^k - \rho_k d^k \\ d^k = \nabla J(x^k) + \frac{\|\nabla J(x^k)\|^2}{\|\nabla J(x^{k-1})\|^2} d^{k-1} \\ \rho_k = \frac{\|\nabla J(x^k)\|^2}{(Ad^k, d^k)} \\ (r^{k+1}, d^k) = 0 \end{cases} \quad (4.13)$$

Il suffit de se donner $d^0 = \nabla J(x^0)$.

N peut être très grand, on peut alors compter le nombre d'opérations nécessaires pour réaliser l'algorithme : une itération nécessite $2cN$ opérations élémentaires, où c est le nombre moyen de coefficients non nuls par ligne de A . Si bien que pour une matrice pleine, le nombre d'opérations élémentaires pour N itérations est $2N^3$. Cela disqualifie la méthode par rapport à Cholewski ($\frac{N^3}{3}$ opérations élémentaires), si on la considère seulement comme une méthode directe.

Remarque 4.5 *On a aussi $\rho_k = -\frac{(r^k, d^k)}{(Ad^k, d^k)}$, ce qui correspond à un gradient à pas optimal dans la direction d^k .*

4.5.3 Analyse de convergence

On introduit l'espace de Krylov

$$\mathcal{K}_k = \text{vect}\{r^0, Ar^0, \dots, A^k r^0\} \quad (4.14)$$

et on a le

Théorème 4.6 . *Si $r^j \neq 0$ pour $j \leq k$, alors $\mathcal{K}_k \equiv \mathcal{L}_k$.*

ON en déduit que $\dim \mathcal{K}_k = k + 1$ et donc que $r^0, Ar^0, \dots, A^k r^0$ forment un système libre. On a donc

$$J(x^{k+1}) = \inf_{v \in u^0 + \mathcal{K}_k} J(v)$$

On en déduit une première estimation de l'erreur

Théorème 4.7

$$E(x^k) = \inf_{P \in \mathbb{P}_{k-1}} \frac{1}{2} (A(I + AP(A))e^0, (I + AP(A))e^0) \quad (4.15)$$

avec $e^0 = u^0 - u$ et $E(v) = \frac{1}{2}(A(v - u), v - u)$.

Par diagonalisation de A on en déduit

Corollaire 4.1

$$E(x^k) = \inf_{P \in \mathbb{P}_{k-1}} \max_{1 \leq i \leq N} [1 + \lambda_i P(\lambda_i)]^2 E(x^0) \quad (4.16)$$

où les λ_i sont les valeurs propres de A .

et par un calcul assez long sur les polynômes de Tchebycheff,

Corollaire 4.2 . On a l'estimation d'erreur

$$E(x^k) \leq 4 \left(\frac{\sqrt{K(A)} - 1}{\sqrt{K(A)} + 1} \right)^{2k} E(x^0) \quad (4.17)$$

De nouveau, la convergence est linéaire. Cette estimation est à comparer avec l'estimation d'erreur (4.9) pour l'algorithme du gradient à pas optimal :

$$E(x^k) \leq \left(\frac{K(A) - 1}{K(A) + 1} \right)^{2k} E(x^0)$$

Par exemple, d'après ces estimations pour $K(A) = 100$, pour obtenir une erreur relative de 10^{-6} sur l'énergie, il faudrait 340 itérations du gradient à pas optimal et seulement 34 itérations du gradient conjugué ! Comme les itérations sont comparables, ces performances font de cet algorithme le favori de tous les gens qui font des calculs de grande taille. De nombreuses extensions ont été proposées : BiCGSTAB, GMRES, etc, pour des problèmes non symétriques, à coefficients complexes, etc..

4.6 Calcul du pas pour les méthodes de descente

Si J n'est pas quadratique, il devient difficile de calculer le pas optimal.

Voir https://optimization.mccormick.northwestern.edu/index.php/Line_search_methods

4.7 Méthodes de Newton et quasi-Newton

On cherche les points critiques de J c'est-à-dire les zéros de $G = J'$. A chaque étape on résout

$$J''(x^k) \cdot (x^{k+1} - x^k) = -J'(x^k).$$

Avantage : on converge plus vite. Inconvénient : on n'est pas sûr de converger, et il faut calculer le Hessien $H(x^k) = J''(x^k)$ et résoudre un système linéaire à chaque étape. On introduit alors les méthodes de quasi-Newton : au lieu de calculer le Hessien à chaque étape, on approche $(H(x^k))^{-1}$ par une matrice W^k qu'on calcule à chaque étape. EN dimension 1, on remplace la méthode de Newton par la méthode de la sécante.