Examen Master SICOM, Cours Optimisation
Luc Pronzato, janvier 2007

QUELQUES INDICATIONS

La partie 1 est totalement indépendante des parties 2 et 3.

On prendra soin de **souligner les vecteurs** et de noter les matrices par des lettres capitales. Un vecteur $\mathbf{v}$ est toujours supposé être du type colonne, on notera $\mathbf{v}^\top$ un vecteur ligne.

ÉNONCÉ DU PROBLÈME

On s'intéresse au comportement d'un algorithme d'optimisation qui utilise une recherche linéaire obéissant à la méthode de descente de Goldstein et Price, rappelée ci-après.

On notera $f(\cdot) : \mathbb{R}^d \to \mathbb{R}$ la fonction à optimiser par rapport à $\mathbf{x} \in \mathbb{R}^d$, $\mathbf{x}_k$ la valeur proposée par l'algorithme d'optimisation à l'itération k. On suppose que $f(\cdot)$ est continûment dérivable sur $\mathbb{R}^d$, on notera $\mathbf{g}(\mathbf{x}) \in \mathbb{R}^d$ son gradient en $\mathbf{x}$ et

$$\mathbf{g}_k = \mathbf{g}(\mathbf{x}_k)\,,\ \ f_k = f(\mathbf{x}_k)\,.$$

La recherche linéaire à l'itération k consiste à trouver une valeur $t_k \in \mathbb{R}$ telle que $f(\mathbf{x}_k + t_k\mathbf{d}_k)$ soit «significativement plus petit» que $f(\mathbf{x}_k)$, avec $\mathbf{d}_k$ la direction de recherche proposée par l'algorithme. On suppose que $\mathbf{d}_k$ est une direction de descente (il existe $t > 0$ tel que $f(\mathbf{x}_k + t\mathbf{d}_k) < f(\mathbf{x}_k)$). On notera

$$h_k(t) = f(\mathbf{x}_k + t\mathbf{d}_k)\,,\ \ h'_k(t) = \frac{dh_k(t)}{dt}$$

(on peut noter que la fonction $h_k(\cdot)$ dépend de $\mathbf{x}_k$ et $\mathbf{d}_k$).

La méthode de Goldstein et Price correspond à l'algorithme ci-dessous.

1. Poser $t_g = 0$, $t_d = \infty$, choisir deux nombres m_1,m_2 tels que $0 < m_1 < m_2 < 1$. Choisir une valeur $t > 0$.
2.
   - (a) Si $h_k(0) + tm_2h'_k(0) \leq h_k(t) \leq h_k(0) + tm_1h'_k(0)$, stop : prendre $t_k = t$.
   - (b) Si $h_k(t) > h_k(0) + tm_1h'_k(0)$, prendre $t_d = t$ (t est trop grand), aller au pas 3.
   - (c) Si $h_k(t) < h_k(0) + tm_2h'_k(0)$, prendre $t_g = t$ (t est trop petit), aller au pas 3.
3. Si $t_d = \infty$, prendre $t = 2t_g$, sinon prendre $t = (t_g + t_d)/2$, retourner au pas 2.

PARTIE 1

1) Expliquer le fonctionnement de la méthode à l'aide d'un dessin représentant l'allure de la fonction $h_k(t)$.

2) On suppose que $h_k(t)$ est quadratique en t avec un minimum en t^*.

- Exprimer $h_k(t^*)$ en fonction de $h_k(0)$, $h'_k(0)$ et t^*.
- Expliquer pourquoi il est souhaitable de choisir $m_1 < 1/2 < m_2$ dans la méthode de Goldstein et Price.
- On pose $\Delta h_k(t^*) = h_k(0) - h_k(t^*)$. Interpréter la partie droite de la condition (a) de la méthode de Goldstein et Price à partir de $\Delta h_k(t)$, la diminution optimale du critère lorsqu'il est quadratique.

On suppose que $f(\cdot)$ est bornée inférieurement et on admet que la méthode de Goldstein et Price retourne alors une valeur t_k satisfaisant la condition (a). Nous allons montrer dans les parties 2 et 3 que, sous des conditions raisonnables, cette méthode de recherche linéaire conduit à un algorithme d'optimisation tel que $\liminf_{k\to\infty} \|\mathbf{g}_k\| = 0$.

PARTIE 2

1) L'algorithme utilise $\mathbf{x}_{k+1} = \mathbf{x}_k + t_k\mathbf{d}_k$. Exprimer $t_kh'_k(0)$ en fonction de $\|\mathbf{g}_k\|$, $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|$ et

$$c_k = -\frac{\mathbf{g}_k^\top\mathbf{d}_k}{\|\mathbf{g}_k\|\,\|\mathbf{d}_k\|}\,.$$

A partir de la partie droite de la condition (a) de la méthode de Goldstein et Price, donner une borne inférieure pour $f_k - f_{k+1}$ en fonction de $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|$ (et aussi $\|\mathbf{g}_k\|$ et c_k).

2) On utilise à présent la partie gauche de la condition (a). Ré-écrire l'inégalité sous la forme d'une borne inférieure pour $(f_{k+1} - f_k)/t_k - \mathbf{g}_k^\top\mathbf{d}_k$ (on exprimera cette borne en fonction de m_2, c_k, $\|\mathbf{g}_k\|$ et $\|\mathbf{d}_k\|$). Utiliser un développement limité de $f(\cdot)$ au premier ordre pour exprimer $(f_{k+1} - f_k)/t_k$ en fonction de $\mathbf{d}_k$ et $\mathbf{g}(\hat{\mathbf{x}}_k)$, avec $\hat{\mathbf{x}}_k$ entre $\mathbf{x}_k$ et $\mathbf{x}_{k+1}$ (c'est-à-dire, $\hat{\mathbf{x}}_k = (1-\alpha)\mathbf{x}_k + \alpha\mathbf{x}_{k+1}$ pour un α dans (0,1)).

3) On suppose que le gradient de $f(\cdot)$ est K-Lipschitzien, c'est-à-dire, qu'il existe $K > 0$ tel que

$$\|\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{z})\| \leq K\|\mathbf{x} - \mathbf{z}\|\ \ \forall\,\mathbf{x}\,,\mathbf{z} \in \mathbb{R}^d\,.$$

En utilisant l'inégalité de Cauchy-Schwarz ($\mathbf{x}^\top \mathbf{z} \le \|\mathbf{x}\|\,\|\mathbf{z}\| \; \forall\, \mathbf{x}\,,\, \mathbf{z} \in \mathbb{R}^d$), déduire de l'inégalité obtenue en 2) une borne inférieure sur $\|\mathbf{x}_{k+1} - \mathbf{x}_k\|$ (en fonction de m_2, c_k, $\|\mathbf{g}_k\|$ et K).

4) Utiliser les résultats de 1) et 3) pour donner une borne inférieure sur $f_k - f_{k+1}$ en fonction de c_k, $\|\mathbf{g}_k\|$ et

$$r = \frac{m_1(1-m_2)}{K}\,.$$

Partie 3

Supposons que la suite des points $\mathbf{x}_k$ proposés par l'algorithme satisfait

$$r c_k^2 \|\mathbf{g}_k\|^2 \le f_k - f_{k+1}$$

avec r constant et $\sum_{k=1}^{\infty} c_k^2 = \infty$.

1) Montrer que la propriété ci-dessus implique que

- soit $f_k \to -\infty$ quand $k \to \infty$,
- soit $\liminf_{k\to\infty} \|\mathbf{g}_k\| = 0$ (c'est-à-dire, $\forall \delta > 0$, $\forall k_0 \ge 1$, $\exists k > k_0$ tel que $\|\mathbf{g}_k\| < \delta$).

2) Déduire de l'inégalité obtenue en 4) de la partie 2 la raison pour laquelle il est primordial de choisir $m_2 < 1$ dans la méthode de Goldstein et Price.

3) On implémente la méthode de Goldstein et Price dans un algorithme tel que

- $\mathbf{d}_k$ est une direction de descente à chaque itération ($c_k > 0$ pour tout k)
- et $\mathbf{d}_k = -\mathbf{g}_k$ pour $k = im$, $i = 1,2,3\ldots$ et m fixé, $m \ge 1$ (c'est-à-dire, la direction de recherche est celle de l'algorithme du gradient toutes les m itérations).

On suppose que la fonction à minimiser est continûment dérivable, de gradient K-Lipschitzien et bornée inférieurement. Que peut on dire de la convergence de l'algorithme ?

4) On l'implémente à présent dans l'algorithme de Newton. On suppose que $f(\cdot)$ est deux fois continûment dérivable, que la matrice $\mathbf{H}(\mathbf{x})$ de ses dérivées secondes est définie positive avec des valeurs propres minimale et maximale $\lambda_{\min}$ et $\lambda_{\max}$ qui satisfont

$$\rho(\mathbf{x}) = \frac{\lambda_{\max}[\mathbf{H}(\mathbf{x})]}{\lambda_{\min}[\mathbf{H}(\mathbf{x})]} \le \bar{\rho} < \infty\,,\ \forall\, \mathbf{x} \in \mathbb{R}^d\,.$$

Donner une borne inférieure pour c_k^2 en fonction de $\rho(\mathbf{x}_k)$ puis de $\bar{\rho}$. On rappelle que d'après l'inégalité de Kantorovich

$$(\mathbf{u}^\top \mathbf{H}^{-1}\mathbf{u})\,(\mathbf{u}^\top \mathbf{H}\mathbf{u}) \le \frac{(\lambda_{\max}(\mathbf{H}) + \lambda_{\min}(\mathbf{H}))^2}{4\lambda_{\max}(\mathbf{H})\lambda_{\min}(\mathbf{H})}\,(\mathbf{u}^\top \mathbf{u})^2\,,\ \forall\, \mathbf{u} \in \mathbb{R}^d\,.$$

Conclure quant à la convergence de l'algorithme.