TP-Projet ModIA
Méthodes numériques pour les problèmes d'optimisation

O.Cots, J. Gergaud, S. Gratton, P. Matalon, C. Royer, D. Ruiz et E. Simon

Année universitaire 2020–2021

Résumé

Ce TP-projet concerne les problèmes d'optimisation sans contraintes. On étudie la méthode de Newton et sa globalisation par l'algorithme des régions de confiance. La résolution du sous-problème des régions de confiance sera réalisée de deux façons, soit à l'aide du point de Cauchy, soit par l'algorithme du Gradient Conjugué Tronqué.

Optimisation sans contrainte

Dans cette partie, on s'intéresse à la résolution du problème

$$\min_{x \in \mathbb{R}^n} f(x),$$

où la fonction f est de classe $\mathcal{C}^2$ sur $\mathbb{R}^n$. On cherche donc à exploiter l'information fournie par ses dérivées première et seconde, que l'on représente en tout point x par le *vecteur gradient* $\nabla f(x) \in \mathbb{R}^n$ et la *matrice Hessienne* $\nabla^2 f(x) \in \mathbb{R}^{n \times n}$.

1 Algorithme de Newton local

Principe

La fonction f étant $\mathcal{C}^2$, on peut remplacer f au voisinage de l'itéré courant x_k par son développement de Taylor au second ordre, soit :

$$f(y) \sim q(y) = f(x_k) + \nabla f(x_k)^T (y - x_k) + \frac{1}{2}(y - x_k)^T \, \nabla^2 f(x_k) \, (y - x_k),$$

On choisit alors comme point x_{k+1} le minimum de la quadratique q lorsqu'il existe et est unique, ce qui n'est le cas que si $\nabla^2 f(x_k)$ est définie positive. Or le minimum de q est réalisé par x_{k+1} solution de : $\nabla q(x_{k+1}) = 0$, soit :

$$\nabla f(x_k) + \nabla^2 f(x_k)(x_{k+1} - x_k) = 0,$$

ou encore, en supposant que $\nabla^2 f(x_k)$ est définie positive :

$$x_{k+1} = x_k - \nabla^2 f(x_k)^{-1} \nabla f(x_k).$$

La méthode ne doit cependant jamais être appliquée en utilisant une inversion de la matrice Hessienne (qui peut être de très grande taille et mal conditionnée), mais plutôt en utilisant :

$$x_{k+1} = x_k + d_k,$$

où d_k est l'unique solution du système linéaire

$$\nabla^2 f(x_k) d_k = -\nabla f(x_k),$$

d_k étant appelée direction de Newton.

Cette méthode est bien définie si à chaque itération, la matrice hessienne $\nabla^2 f(x_k)$ est définie positive : ceci est vrai en particulier au voisinage de la solution x^* cherchée si on suppose que $\nabla^2 f(x^*)$ est définie positive (par continuité de $\nabla^2 f$).

1.1 Algorithme

Algorithme 1 ALGORITHME DE NEWTON (LOCAL)

Données : f , x_0 première approximation de la solution cherchée, $\epsilon > 0$ précision demandée.

Sortie : une approximation de la solution du problème $\min_{x \in \mathbb{R}^n} f(x)$.

1.Tant que le test de convergence est non satisfait :

a. Calculer d_k solution du système : $\nabla^2 f(x_k) \, dk = -\nabla f(x_k)$,

b. Mise à jour : $x_{k+1} = x_k + d_k$, $k = k + 1$,

2. Retourner x_k.

2 Régions de confiance - Partie 1

L'introduction d'une *région de confiance* dans la méthode de Newton permet de garantir la convergence globale de celle-ci, i.e. la convergence vers un optimum local quel que soit le point de départ. Cela suppose certaines conditions sur la résolution locale des sous-problèmes issus de la méthode, qui sont aisément imposables.

Principe

L'idée de la méthode des régions de confiance est d'approcher f par une fonction modèle plus simple m_k dans une région $R_k = \{x_k + s; \|s\| \leq \Delta_k\}$ pour un Δ_k fixé.

Cette région dite "de confiance" doit être suffisament petite pour que

$$m_k(x_k + s) \sim f(x_k + s).$$

Le principe est que, au lieu de résoudre l'équation : $f(x_{k+1}) = \min_{\|s\| \leq \Delta_k} f(x_k + s)$, on résout :

$$m_k(x_{k+1}) = \min_{\|s\| \leq \Delta_k} m_k(x_k + s) \tag{2.1}$$

Si la différence entre $f(x_{k+1})$ et $m_k(x_{k+1})$ est trop grande, on diminue le Δ_k (et donc la région de confiance) et on résout le modèle (2.1) à nouveau. Un avantage de cette méthode est que toutes les directions sont prises en compte. Par contre, il faut faire attention à ne pas trop s'éloigner de x_k ; en général, la fonction m_k n'approche proprement f que sur une région proche de x_k.

Exemple de modèle : l'approximation de Taylor à l'ordre 2 (modèle quadratique) :

$$m_k(x_k + s) = q_k(s) = f(x_k) + g_k^\top s + \frac{1}{2} s^\top H_k s \tag{2.2}$$

avec $g_k = \nabla f(x_k)$ et $H_k = \nabla^2 f(x_k)$.

2.1 Algorithme

Algorithme 2 MÉTHODE DES RÉGIONS DE CONFIANCE (ALGO GÉNÉRAL)

Données : $\Delta_{max} > 0$, $\Delta_0 \in (0, \Delta_{max})$, $0 < \gamma_1 < 1 < \gamma_2$ et $0 < \eta_1 < \eta_2 < 1$.

Sortie : une approximation de la solution du problème : $\min_{x \in \mathbb{R}^n} f(x)$.

1.Tant que le test de convergence n'est pas satisfait :

a. Calculer approximativement s_k solution du sous-problème (2.1) ;

b. Evaluer $f(x_k + s_k)$ et $\rho_k = \dfrac{f(x_k) - f(x_k + s_k)}{m_k(x_k) - m_k(x_k + s_k)}$

c. Mettre à jour l'itéré courant :

$$x_{k+1} = \begin{cases} x_k + s_k & \text{si } \rho_k \geq \eta_1 \\ x_k & \text{sinon.} \end{cases}$$

d. Mettre à jour la région de confiance :

$$\Delta_{k+1} = \begin{cases} \min\{\gamma_2 \Delta_k, \Delta_{\max}\} & \text{si } \rho_k \geq \eta_2 \\ \Delta_k & \text{si } \rho_k \in [\eta_1, \eta_2) \\ \gamma_1 \Delta_k & \text{sinon.} \end{cases}$$

2. Retourner x_k.

L'algorithme 2 est un cadre générique. On va s'intéresser à deux raffinages possibles de l'étape a.

2.2 Le pas de Cauchy

On considère ici le modèle quadratique $q_k(s)$. Le sous-problème de régions de confiance correspondant peut se révéler difficile à résoudre (parfois autant que le problème de départ). Il est donc intéressant de se restreindre à une résolution approchée de ce problème.

Le pas de Cauchy appartient à la catégorie des solutions approchées. Il s'agit de se restreindre au sous-espace engendré par le vecteur g_k ; le sous-problème s'écrit alors

$$\left\{ \begin{array}{ll} \min & q_k(s) \\ s.t. & s = -t\, g_k \\ & t > 0 \\ & \|s\| \leq \Delta_k. \end{array} \right. \tag{2.3}$$

3 Régions de confiance - Partie 2

Dans la section précédente, on a pu voir que la technique du pas de Cauchy ne garantit pas une convergence rapide en général ; on retrouve ici le problème d'une méthode de descente de gradient. On souhaite donc étudier une méthode pour la résolution approchée du sous-problème avec région de confiance (2.1), qui puisse récupérer asymptotiquement la convergence quadratique inhérente à la méthode de Newton Local. L'algorithme du Gradient Conjugué Tronqué appartient à cette catégorie.

3.1 Algorithme du Gradient Conjugué Tronqué

On s'intéresse maintenant à la résolution approchée du problème (2.1) à l'itération k de l'algorithme 2 des Régions de Confiance. On considère pour cela l'algorithme du Gradient Conjugué Tronqué :

Algorithme 3 ALGORITHME DU GRADIENT CONJUGUÉ TRONQUÉ

Données : $\Delta_k > 0$, x_k, $g = \nabla f(x_k)$, $H = \nabla^2 f(x_k)$.

Sortie : le pas s qui approche la solution du problème : $\min_{\|s\| \leq \Delta_k} q(s)$
où $q(s) = g^\top s + \frac{1}{2} s^\top H_k\, s$.

Initialisations : $s_0 = 0$, $g_0 = g$, $p_0 = -g$;

1. Pour $j = 0, 1, 2, \ldots$, **faire :**

a. $\kappa_j = p_j^T H p_j$

b. Si $\kappa_j \leq 0$, alors
  déterminer σ_j la racine de l'équation $\|s_j + \sigma p_j\|_2 = \Delta_k$
   pour laquelle la valeur de $q(s_j + \sigma p_j)$ est la plus petite.
  Poser $s = s_j + \sigma_j p_j$ et sortir de la boucle.
 Fin Si

c. $\alpha_j = g_j^T g_j / \kappa_j$

d. Si $\|s_j + \alpha_j p_j\|_2 \geq \Delta_k$, alors
  déterminer σ_j la racine positive de l'équation $\|s_j + \sigma p_j\|_2 = \Delta_k$.
  Poser $s = s_j + \sigma_j p_j$ et sortir de la boucle.
 Fin Si

e. $s_{j+1} = s_j + \alpha_j p_j$

f. $g_{j+1} = g_j + \alpha_j H p_j$

g. $\beta_j = g_{j+1}^T g_{j+1} / g_j^T g_j$

h. $p_{j+1} = -g_{j+1} + \beta_j p_j$

i. Si la convergence est suffisante, poser $s = s_{j+1}$ et sortir de la boucle.

2. Retourner s.

A Problèmes Tests

Les problèmes de minimisation sans contraintes à résoudre sont les suivants :

Problème 1

$$\begin{aligned} f_1 : \quad \mathbb{R}^3 & \rightarrow \mathbb{R} \\ (x_1, x_2, x_3) & \mapsto 2\,(x_1 + x_2 + x_3 - 3)^2 + (x_1 - x_2)^2 + (x_2 - x_3)^2. \end{aligned}$$

On cherchera à minimiser f_1 sur $\mathbb{R}^3$, en partant des points suivants

$$x_{011} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \quad x_{012} = \begin{bmatrix} 10 \\ 3 \\ -2.2 \end{bmatrix}.$$

Problème 2

$$\begin{aligned} f_2 : \quad \mathbb{R}^2 & \rightarrow \mathbb{R} \\ (x_1, x_2) & \mapsto 100\,(x_2 - x_1^2)^2 + (1 - x_1)^2. \end{aligned}$$

On cherchera à minimiser f_2 sur $\mathbb{R}^2$, en partant des points suivants

$$x_{021} = \begin{bmatrix} -1.2 \\ 1 \end{bmatrix}, \quad x_{022} = \begin{bmatrix} 10 \\ 0 \end{bmatrix}, \quad x_{023} = \begin{bmatrix} 0 \\ \frac{1}{200} + \frac{1}{10^{12}} \end{bmatrix}.$$

B Cas tests pour le calcul du pas de Cauchy

On considère des fonctions quadratiques de la forme $q(s) = s^\top g + \frac{1}{2} s^\top H\, s$.

Quadratique 1

$$g = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad H = \begin{bmatrix} 7 & 0 \\ 0 & 2 \end{bmatrix}.$$

Quadratique 2

$$g = \begin{bmatrix} 6 \\ 2 \end{bmatrix}, \quad H = \begin{bmatrix} 7 & 0 \\ 0 & 2 \end{bmatrix}.$$

Quadratique 3

$$g = \begin{bmatrix} -2 \\ 1 \end{bmatrix}, \quad H = \begin{bmatrix} -2 & 0 \\ 0 & 10 \end{bmatrix}.$$

C Cas tests pour la résolution du sous-problème par l'algorithme du Gradient Conjugué Tronqué

On reprendra les 3 quadratiques testées avec le pas de Cauchy, auxquelles on ajoutera :

Quadratique 4

$$g = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad H = \begin{bmatrix} -2 & 0 \\ 0 & 10 \end{bmatrix}.$$

Quadratique 5

$$g = \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \quad H = \begin{bmatrix} 4 & 6 \\ 6 & 5 \end{bmatrix}.$$

Quadratique 6

$$g = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \quad H = \begin{bmatrix} 4 & 0 \\ 0 & -15 \end{bmatrix}.$$