UNIVERSITÉ DE FRIBOURG SUISSE
UNIVERSITÄT FREIBURG SCHWEIZ

Otimisation numérique 3

Méthode de recherche par ligne
(chapitre 3.3 et 3.4 du livre)

Chiara Piccioli

Proséminaire
15 Octobre 2009

Table des matières

1 Taux de convergence

1.1 Introduction

Définition 1. Un algorithme *converge globalement* ssi $\lim_{n\to\infty} ||\nabla f(x_k)|| = 0$.

Soit $\cos\Phi_k = \frac{-\nabla f_k^T p_k}{||\nabla f_k|| ||p_k||}$. Par la condition de Zoutendijk, pour la convergence globale, on doit avoir $\cos^2\Phi_k ||\nabla f_k|| \to 0$ pour $k \to \infty$ A partir de cette conclusion, définir des algorithmes avec des bonnes propriétés de convergence peut paraître simple, car tous ce qu'il faut c'est d'assurer que p_k ne devient pas orthogonale à ∇f_k.
On peut donc simplement calculer $\cos\phi_k$ à chaque itération et tourner p_k vers la direction de la plus grande pente si cos est plus petit qu'une certaine constante $\delta > 0$. Un test de ce genre assure une convergence globale, mais il n'est pas reccomandé pour deux raisons :

1. il empêche une convergence rapide
2. les tests sur les angles detruisent les propriétés invariables des méthodes quasi-Newton.

Le but est donc de construire des algorithmes qui ont une bonne convergence globale et un bon taux de convergence. On commence notre étude sur les taux de convergence avec des méthodes de recherche linéaires : la méthode de la plus grande pente.

1.2 Taux de convergence de la plus grande pente

On considère le cas idéal où la fonction est quadratique et les lignes de recherche sont exactes. On suppose que

$$f(x) = \frac{1}{2}x^T Q x - b^T x \tag{1}$$

où Q est symétrique et positive. Le gradient est donné par $\nabla f(x) = Qx - b$ et x^* est la seule solution de $\nabla f(x) = 0$. On veut maintenant trouver la longeur du pas α_k qui minimalise $f(x_k - \alpha\nabla f_k)$, simplement en dérivant cette dernière par rapport à α. On obtient donc

$$\alpha_k = \frac{\nabla f_k^T \nabla f_k}{\nabla f_k^T Q \nabla f_k} \tag{2}$$

En utilisant α_k, l'iteration de la plus grande pente est donnée par

$$x_{k+1} = x_k - \frac{\nabla f_k^T \nabla f_k}{\nabla f_k^T Q \nabla f_k}\nabla f_k \tag{3}$$

Ce qu'on veut faire maintenant, c'est de quantifier le taux de convergence. Pour faire ceci on a besoin de la norme $||x||_Q^2 = x^T Q x$. À l'aide de l'équation $Qx^* = b$ on peut tout simplement montrer que

$$\frac{1}{2}||x - x^*||_Q^2 = f(x) - f(x^*) \tag{4}$$

Cette norme mesure donc la différence qu'il y a entre la valeur courrante de f et la valeur optimale. En utilisant la formule (3) et le fait que $\nabla f_k = Q(x_k - x^*)$, on peut dériver l'égalité

$$||x_{k+1} - x^*||_Q^2 = \{1 - \frac{(\nabla f_k^T \nabla f_k)^2}{(\nabla f_k^T Q \nabla f_k)(\nabla f_k^T Q^{-1} \nabla f_k)}\}||x_k - x^*||_Q^2 \tag{5}$$

Cette expression est interessante pour voir l'exacte diminution de f à chaque iteration, mais on peut bien s'imaginer que la parantèse est difficile à interpreter. Pour cette raison, il est mieux de se referer en termes de nombre de condition du problème.

Théorème 1. *Si la méthode de la plus grande pente avec des lignes de recherche exactes (3) est appliqué à la fonction quadratique convexe (1) la norme de l'erreur satisfait*

$$||x_{k+1} - x^*||_Q^2 \leq (\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1})^2 ||x_k - x^*||_Q^2 \tag{6}$$

avec $0 < \lambda_1 \leq ... \leq \lambda_n$ qui sont les valeurs propres de Q.

On peut donc voir par cette inégalité et par la formule précédente (4) que les valeurs de f_k convergent linéairement vers f^*.
Si on prends le cas spécial où les valeurs propres sont tous les mêmes, on aura que Q est un multiple de la matrice identité et ldonc les lignes de niveau de $f(x)$ ne sont plus des ellipse mais des cercles, et de plus la direction de la plus grande pente converge vers la solution

Théorème 2. *On suppose que $f : \mathbb{R}^n \to \mathbb{R}$ soit deux fois continuement differentiable et que les iterés generés par la méthode de la plus grande pente avec des recherches linéaires exactes convergent vers x^*, où $\nabla^2 f(x^*)$ est définie positif. Soit r un scalaire qui satisfait la condition $r \in (\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}, 1)$ où $\lambda_1 \leq ... \leq \lambda_n$ sont des valeurs propres de $\nabla^2 f(x^*)$. Alors pour chaque k suffisamment grand, on a*

$$f(x_{k+1}) - f(x^*) \leq r^2[f(x_k) - f(x^*)].$$

En général on ne peut pas s'attendre que le taux de convergence devient meilleur si on utilise une recherche linéaire inexacte. Ce théorème montre donc que la méthode de la plus grande pente peut avoir un taux de convergence trop bas, même quand $\nabla^2 f(x^*)$ est bien conditionné.

Exemple 1. Si $k(Q) = 800$, $f(x_1) = 1$,et $f(x^*) = 0$ alors on aura que la valeurs de la fonction sera environ 0.08 après 1000 iteration de la méthode de la plus grande pente.

1.3 Méthode quasi-Newton

On doit tout d'abord supposer que

$$p_k = -B_k^{-1} \nabla f_k \tag{7}$$

où B_k est une matrice symétrique positive. On suppose de plus que le pas de longeur α_k est calculé depuis une ligne de recherche inexacte qui satisfait la condition de Wolfe, avec une particularité : la recherche linéaire de l'algorithme essaye toujours d'abord avec $\alpha_k = 1$ et acceptera cette valeur de α_k, si elle satisfait les conditions de Wolfe. Ceci est naturellement fait pour aumenter le taux de convergence. Le résultat suivant nous montre que si la direction de recherche d'une méthode quasi-Newton satisfait la direction de Newton, alors la longeur du pas $\alpha_k = 1$ satisfait la condition de Wolfe quand les itérés convergent vers la solution.

Théorème 3. *Soit $f : \mathbb{R}^n \to \mathbb{R}$ 2 fois continûment differentiable. On considère l'itération $x_{k+1} = x_k + \alpha_k p_k$, où p_k est la direction de pente et α_k satisfait la condition de Wolfe avec $c_1 \leq \frac{1}{2}$. Si la suite $\{x_k\}$ converge vers x^*, tel que $\nabla f(x^*) = 0$ et que $\nabla^2 f(x^*)$ soit défini positif, et si*

$$\lim_{k \to \infty} \frac{||\nabla f_k + \nabla^2 f_k p_k||}{||p_k||} = 0, \tag{8}$$

alors :

1. *il existe k_0 tel que $\alpha_k = 1$ est accetable pour chaque $k > k_0$*
2. *si $\alpha_k = 1 \forall k > k_0$, $\{x_k\}$ converge superlinéairement vers x^*.*

On peut constater que si p_k est une direction de recherche de quasi-Newton de la forme (7), alors (8) se laisse écrire comme

$$\lim_{k \to \infty} \frac{||(B_k + \nabla^2 f(x^*))p_k||}{||p_k||} = 0 \tag{9}$$

Ce pas est trés importante parce que nous permet de voir que si B_k ne converge pas vers $\nabla^2 f(x^*)$, une convergence superlinéaire peut arriver comme même ; il faut seulement que les B_k deviennent des bonnes approximations de $\nabla^2 f(x^*)$ dans la direction de p_k.

Théorème 4. *Supposons que $f : \mathbb{R}^n \to \mathbb{R}$ 2 fois continument differentiable. On considère l'itération $x_{k+1} = x_k + p_k$ (donc la longeur du pas est toujours 1) et que p_k est donné par $p_k = -B_k^{-1} \nabla f_k$. On suppose même que $\{x_k\}$ converge vers un point x^* tel que $\nabla f(x^*) = 0$ et $\nabla^2 f(x^*)$ est defini positif. Alors $\{x_k\}$ converge superlinéairement si et seulement si* (9) *est vrai.*

Démonstration. On doit montrer tout d'abord que (9) est égal à

$$p_k - p_k^N = o(||p_k||)$$

où $p_k^N = -\nabla^2 f_k^{-1} \nabla f_k$ est le pas de Newton. Donc,

$$\begin{aligned} p_k - p_k^N &= \nabla^2 f_k^{-1}(\nabla^2 f_k p_k + \nabla f_k) \\ &= \nabla^2 f_k^{-1}(\nabla^2 f_k - B_k)p_k \\ &= O(||(\nabla^2 f_k - B_k)p_k||) \\ &= o(||p_k||) \end{aligned}$$

Ici on a utilisé le fait que $||\nabla^2 f_k^{-1}||$ est borné pour des x_k suffisamment proches de x^*, parce que l'Hessien $\nabla^2 f(x^*)$ est défini positif. L'inverse suive directement si on multiplie par $\nabla^2 f_k$ les deux cotés de $p_k - p_k^N = o(||p_k||)$. Pour la suite de la preuve on doit regarder la preuve du théorème 5, et en particulier on doit utiliser l'inégalité

$$||x_k + p_k^N - x^*|| \leq L||\nabla^2 f(x^*)^{-1}||||x_k - x^*||^2 = \tilde{L}||x_k - x^*||^2.$$

A l'aide de $p_k - p_k^N = o(||p_k||)$.

On obtient donc que

$$||x_k + p_k - x^*|| \leq ||x_k + p_k^N - x^*|| + ||p_k - p_k^N|| = O(||x_k - x^*||^2) + o(||p_k||)$$

Avec une simple manipulation de cette inégalité on obtient que $||p_k|| = O(||x_k - x^*||)$, et donc on obtient que

$$||x_k + p_k - x^*|| \leq o(||x_k - x^*||)$$

qui donne le resultat de convergence superlinéaire. □

1.4 Méthode de Newton

Dans ce cas on considère l'itération de Newton, où

$$p_k^N = -\nabla^2 f_k^{-1} \nabla f_k \tag{10}$$

À cause du fait que $\nabla^2 f_k$ peut ne pas être défini positif, p_k^N n'est pas nécessairement une direction de descente. Le théorème 3 ne pourra donc pas être appliquée aux idées vue dans ce chapitre.

Maintenant on veut discuter les propriétés des taux de convergence locales. Supposons que $\nabla^2 f(x)$ soit lipschittzien autours de x^*. Le $\nabla^2 f(x)$ est alors defini positif dans un voisinage de x^* et donc la méthode de Newton sera donc bien définie dans cette region et convergera quadratiquement a condition que le pas de longeur α_k soit toujours 1.

Théorème 5. *On suppose que f soit 2 fois continûment différentiable et que $\nabla^2 f(x)$ est lipschitzienne et continue dans les environs de la solution x^* dans laquelle les conditions suffisantes (Thm 2.4 du livre) sont satisfaites. On considère l'itération $x_{k+1} = x_k + p_k^N$ où p_k^N est donné par (10). Alors*

1. *si le point de départ x_0 est suffisamment proche de la solution, $\{x_k\}$ converge vers x^**
2. *le taux de convergence de $\{x_k\}$ est quadratique*
3. *$\{||\nabla f_k||\}$ converge quadratiquement vers 0.*

Démonstration. A partir de la définition du pas de Newton et la condition d'optimilité $\nabla f_* = 0$ on a que

$$x_k + p_k^N - x^* = x_k - x^* - \nabla^2 f_k^{-1} \nabla f_k$$

$$= \nabla^2 f_k^{-1} [\nabla^2 f_k (x_k - x^*) - (\nabla f_k - \nabla f_*)]$$

Car

$$\nabla f_k - \nabla f_* = \int_0^1 \nabla^2 f(x_k + t(x^* - x_k))(x_k - x^*) dt$$

on a

$$\begin{aligned} ||\nabla^2 f(x_k)(x_k - x^*) - (\nabla f_k - \nabla f(x^*))|| &= ||\int_0^1 [\nabla^2 f(x_k) - \nabla^2 f(x_k + t(x^* - x_k))](x_k - x^*) dt|| \\ &\leq \int_0^1 ||\nabla^2 f(x_k) - \nabla^2 f(x_k + t(x^* - x_k))|| \cdot ||x_k - x^*|| dt \\ &\leq ||x_k - x^*||^2 \int_0^1 Lt dt \\ &= \frac{1}{2} L ||x_k - x^*||^2, \end{aligned}$$

où L est une constant de Lipschitz pour $\nabla^2 f(x)$ pour x proche de x^*. A cause du fait que $\nabla^2 f(x^*)$ est non singulaire, il existe un rayon $r > 0$ tel que $[\nabla^2 f_k]^{-1}$ existe et que $||\nabla^2 f_k^{-1}|| \leq 2||\nabla^2 f(x^*)^{-1}||$ pour tout x_k avec $||x_k - x^*|| \leq r$. On obtient

$$||x_k + p_k^N - x^*|| \leq L ||\nabla^2 f(x^*)^{-1}|| \cdot ||x_k - x^*||^2 = \tilde{L} ||x_k - x^*||^2$$

avec $\tilde{L} = L||\nabla^2 f(x^*)^{-1}||$. En choisissant x_0 tel que $||x_0 - x^*|| \leq \min(r, 1/(2L))$ on peut utiliser cette inégalité inductive pour déduire que la suite $\{x_k\}$ converge vers x^*, et que le taux de convergence est quadratique. En utilisant la rélation $x_{k+1} - x_k = p_k^N$ et $\nabla f_k + \nabla^2 f_k p_k^N = 0$ on obtient que

$$
\begin{aligned}
||\nabla f(x_{k+1})|| &= ||\nabla f(x_{k+1} - \nabla f_k - \nabla^2 f(x_k) p_k^N|| \\
&= || \int_0^1 \nabla^2 f(x_k + t p_k^N)(x_{k+1} - x_k) dt - \nabla^2 f(x_k) p_k^N|| \\
&\leq \int_0^1 ||\nabla^2 f(x_k + t p_k^N) - \nabla^2 f(x_k)|| \cdot ||p_k^N|| dt \\
&\leq \frac{1}{2} L ||p_k^N||^2 \\
&\leq \frac{1}{2} L ||\nabla^2 f(x_k)^{-1}||^2 ||\nabla f_k||^2 \\
&\leq 2L ||\nabla^2 f(x^*)^{-1}||^2 ||\nabla f_k||^2,
\end{aligned}
$$

qui prouve que le gradient converge quadratiquement vers 0. □

1.5 Méthode de déscente coordonnée par coordonnée

Cette méthode consiste en utiliser cycliquement les n directions coordonnés $e_1, e_2, ..., e_n$ comme direction de recherche.
À l'itération k on cherche α_k qui minimalise $f(X_K + \alpha_k e_k)$. Après n itération on retrouve la première variable et on repète le cycle. Dans la pratique cette méthode peut être inefficient pour une fonction quadratique de deux variables.

Un grand problème de cette méthode, c'est que cette méthode peut itérer infiniment sans jamais aborder un point où le gradient de la fonction disparait. En plus, une recherche cyclique ne garantit pas une convergence globale. La difficulté consiste dans le fait que le gradient peut devenir toujours plus perpendiculaire aux cordonnées de direction de recherche, et donc le cosinus de l'angle devient toujours plus proche de 0. Il peut alors satisfaire é la condition de Zoutendijk sans que le grandient tende vers 0. On peut encore constater que si cette méthode converge vers une solution, d'habitude elle converge plus lentement que la méthode de la plus grande pente.

Même si cette méthode possède tous ces défauts, elle peut être encore utile pour deux raison :

1. on ne doit pas calculer le gradient de f,
2. la vitesse de convergence peut être suffisamment acceptable si les variables sont faiblement dépendants, càd si $\nabla^2 f(x)$ est presque diagonal.

2 Algorithme pour le choix de la longeur du pas

9 On veut maintenant considérer les techniques pour trouver un minimum de la fonction

$$\phi(\alpha) = f(x_k + \alpha p_k) \tag{11}$$

ou bien pour trouver la longeur du pas α_k qui satisfait des conditions de Wolfe. On suppose que p_k soit une direction de déscente, ce qui limite la recherche à des valeurs de α positives. Si f est une fonction quadratique $f(x) = \frac{1}{2}x^T Qx + b^T x + c$ alors son minimum, le long du rayon $x_k + \alpha p_k$ est donné par

$$\alpha_k = -\frac{\nabla f_k^T p_k}{p_k^T Q p_k}. \tag{12}$$

Pour les fonctions non linéaires, il est nécessaire d'utiliser une méthode iterative pour calculer α_k. Le choix d'une α_k acceptable a un impact majeur sur la robustesse et l'efficacité des méthodes d'optimisation non linéaires.
Le pas α_k est cherché sur les lignes données par la direction de déscente p_k. D'où le nom de cette classe de méthodes " de recherche par ligne".

Une recherche demande une valeur initiale α_0, et elle genère une séquence $\{\alpha_i\}$ soit converge vers un α qui satisfait les conditions de Wolfe, soit elle determine que cette longeur n'existe pas. Une procedure typique consiste en deux phases : une "bracketing phase" qui trouve un intervalle $[a, b]$ qui contient des α acceptables, et une phase de sélection qui localise la longeur du pas finale. La phase de séléction d'habitude réduit l'intervalle, en interpolant des informations quôn obtient dans les intérations précendants.

2.1 Interpolation

Le but est de trouver une valeur de α qui satisfait la condition de Wolfe

$$f(x_k + \alpha_k p_k) \leq f(x_k) + c_1 \alpha_k \nabla f_k^T p_k.$$

On aura donc une séquence de valeurs decroissantes α_i tel que la valuer α_i ne soit pas trop plus petit de α_{i-1}. Si ceci arrive, on calcule $\alpha_1 = \alpha_{i-1}/2$. Ceci nous permet de faire des progress à chaque iteration et que l'α finale ne soit trop petit. On peut constater qu'on peut écrire la condition de Wolfe comme

$$\phi(\alpha_k) \leq \phi(0) + c_1 \alpha_k \phi'(0) \tag{13}$$

On propose un algorithme de recherche des α_k de façon qu'il calcule $\nabla f(x)$ le moins de fois possible. On suppose que α_0 est donnée. Si on a que

$$\phi(\alpha_0) \leq \phi(0) + c_1 \alpha_0 \phi'(0)$$

alors la recherche est terminé. Autrement, on sait que l'intervalle $[0, \alpha_0]$ contient des longeur acceptables. Le pas suivant consiste en approximer ϕ par $\phi_q(\alpha)$, en interpolant $\phi(0)$, $\phi'(0)$ et $\phi(\alpha_0)$ pour obtenir :

$$\phi_q(\alpha) = \left(\frac{\phi(\alpha_0) - \phi(0) - \alpha_0 \phi'(0)}{\alpha_0^2} \right) \alpha^2 + \phi'(0)\alpha + \phi(0) \tag{14}$$

Le pas α_0 étant rejété, on cherche le candidat suivant, α_1, pour minimaliser approximativement $\phi(\alpha)$. En fait, au lieu de minimaliser $\phi(\alpha)$ (qui est en général non-linéaire), on minimalise l'interpolant $\phi_q(\alpha)$. Le minimum est atteint dans

$$\alpha_1 = -\frac{\phi'(0)\alpha_0^2}{2[\phi(\alpha_0) - \phi(0) - \phi'(0)\alpha_0]} \tag{15}$$

qui est le minimiseur de notre fonction quadratique $\phi_q(\alpha)$. Si la condition (13) est satisfait, notre recherche est terminé. Autrement en continue en interpolant les valeurs de $\phi(0)$, $\phi'(0)$ et deux valeurs les plus recents de ϕ (pour le prochaine passage ça serait $\phi(\alpha_0)$ et $\phi(\alpha_1)$).

2.2 La longeur du pas initial

Pour la méthode de Newton et de quasi-Newton on commence toujours avec la longeur du pas initial $\alpha_0 = 1$. Par contre, pour les méthodes qui ne produisent pas un pas de longeur optimal, on doit utiliser les informations du problème et les algorithmes pour trouver la longeur α_0. Une stratégie populaire est de choisir α_0 tel que $\alpha_0 \nabla f_k^T p_k = \alpha_{k-1} \nabla f_{k-1}^T p_{k-1}$. On obtient donc que

$$\alpha_0 = \alpha_{k-1} \frac{\nabla f_{k-1}^T p_{k-1}}{\nabla f_k^T p_k}$$

Une autre stratégie est d'interpoler $f(x_{k-1})$, $f(x_k)$ et $\phi'(0) = \nabla f_{k-1}^T p_{k-1}$ pour un polynôme de degré 3 et de définir α_0 comme le minimaliseur de l'interpolant.On obtient ainsi

$$\alpha_0 = \frac{2(f_k - f_{k-1})}{\phi'(0)} \tag{16}$$

On peut observer que si x_k converge superlinéairement vers x^*, alors la fraction (16) converge vers 1. Si on remplace dans (16) $\alpha_0 \leftarrow \min(1, 1.01\alpha_0)$, on trouve qu'on essaie et on accepte toujours $\alpha_0 = 1$, et on peut alors observer les propriétés de la convergence superlinéaire des méthodes de Newton et quasi-Newton.

2.3 Algorithme de recherche pour la condition de Wolfe

Dans cette section on veut décrire une procedure qui nous permet de trouver la longeur du pas pour la condition forte de Wolfe. D'abord on doit supposer que p est une direction de descente et que f est borné d'en bas le long de la direction p. L'algorithme qui nous permet de trouver α, est divisé en deux phases. Dans la première phase on estime α_1, et on l'incremente jusqu'à on trouve une longeur acceptable ou un intervalle qui contient la longeur desirée. La deuxième phase, par contre, est caractérisée par la fonction zoom, qui diminue la taille de l'intervalle jusqu'à ce qu'on converge vers un α_k qui vérifie les conditions de Wolfe.

Algoritme 1(Line Search Algorithm)
Set $a_0 \leftarrow 0$, choose $\alpha_1 > 0$ and α_{max}
$i \leftarrow 1$;
repeat
Evaluate $\Phi(\alpha_i)$;
if $\Phi(\alpha_i) > \Phi(0) + c_1\Phi'(0)$ or $[\Phi(\alpha_i) \geq \Phi(\alpha_{i-1})$ and $i > 1]$
$\alpha_* \leftarrow zoom(\alpha_{i-1}, \alpha_i)$ and stop ;
Evaluate $\Phi'(\alpha_i)$;
if $|\Phi'(\alpha_i)| \leq -c_2\Phi'(0)$
set $\alpha_* \leftarrow \alpha_i$ and stop ;
if $\Phi'(\alpha_i) \geq 0$
set $\alpha_* \leftarrow zoom(\alpha_i, \alpha_{i-1})$ and stop ;
Choose $\alpha_{i+1} \in (\alpha_i, \alpha_{max})$ and stop ;
$i \leftarrow i + 1$
end(repeat)

Cette alorithme utilise le fait que l'intervalle (α_{i-1}, α_i) contient des longeur qui satisfont la condition forte de Wolfe si une des trois conditions est satisfait :

1. α_i ne satisfait pas la condition de Wolfe
2. $\phi(\alpha_i) \geq \phi(\alpha_{i-1})$
3. $\phi'(\alpha_i) \geq 0$

Le dernier passage de l'algorithme nous permet de trouver la prochaine valeur α_{i+1}. Pour incrementer ce pas, on peut utiliser différents méthodes, comme l'interpolation ou simplement on peut multiplier α_i par une constante plus grande que 1. La seule chose importante est que la suite des pas $\{\alpha_1\}$augmente suffisamment vite pour arriver à α_{max} en un nombre raisonnable d'opérations.
Maintenant on veut voir plus en détail la fonction *zoom*. L'ordre de ses inputs est telle que chaque evaluation possède la forme $zoom(\alpha_{lo}, \alpha_{hi})$ où :

1. $[\alpha_{lo}; \alpha_{hi}]$ ou $[\alpha_{hi}; \alpha_{lo}]$ contient une longeur du pas qui satisfait la condition forte de Wolfe ;
2. $\phi(\alpha_l o) < \phi(\alpha_{hi})$
3. α_{hi} est choisi tel que $\phi'(\alpha_{lo})(\alpha_{hi} - \alpha_{lo}) < 0$

Chaque itération de la fonction *zoom* produit un α_j entre α_{lo} et α_{hi} et après il remplace un des deux points extrêmes de l'interval par α_j : si α_j satisfait la condition de diminution suffisante et possède une valeur de la fonction inferieur à α_{lo}, alors on remplace α_{lo} par α_j.

Algoritme 2$(zoom(\alpha_{lo}, \alpha_{hi}))$

repeat

  Interpolate (using quadratic, cubic, or bisection) to find a trial step lenght α_j between α_{lo} and α_{hi} ;

  Evaluate $\Phi(\alpha_j)$;

  if $\Phi(\alpha_j) > \Phi(0) + c_1\alpha_j\Phi'(0)$ ou $\Phi(\alpha_j) \geq \Phi(\alpha_{lo})$

    $\alpha_{hi} \leftarrow \alpha_j$;

  else

    Evaluate $\Phi'(\alpha_j)$

    if $|\Phi'(\alpha_j)| \leq -c_2\Phi'(0)$

      Set $\alpha_* \leftarrow \alpha_j$ and stop ;

    if $\phi'(\alpha_j)(\alpha_{hi} - \alpha_{lo}) \geq 0$

      $\alpha_{hi} \leftarrow \alpha_{lo}$

    $\alpha_{lo} \leftarrow \alpha_j$

end(repeat)