

Analyse des deux méthodes de Broyden

Patrick Roux, 6^e semestre de mathématiques

20 juin 2002

Introduction

Les méthodes de Broyden sont deux algorithmes permettant de résoudre un système de n équations non-linéaires données par une fonction $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$.

Or, il se trouve que dans la littérature traitant le sujet, une de ces méthodes est systématiquement appelée la “bonne” méthode de Broyden et l’autre la “mauvaise”. Toutefois, il n’a jamais été démontré que la “bonne” méthode fonctionne réellement mieux que la “mauvaise”.

Le but de ce rapport est donc de répondre aux questions suivantes :

- La “bonne” méthode fonctionne-t-elle toujours mieux que la “mauvaise” ?
- Si c’est le cas, pourquoi ?
- Si ce n’est pas le cas, dans quelles situations la “mauvaise” est-elle plus performante que la “bonne” ?

De par la similarité des deux méthodes, il semble curieux qu’une d’elles fonctionne toujours mieux que l’autre. Les tests réalisés auparavant étaient peut-être toujours mieux adaptés à la “bonne” méthode ...

C’est cette situation un peu confuse que je vais tenter de clarifier dans ce rapport.

Méthodes de Broyden

Pour commencer, décrivons ces méthodes.

Comme décrit dans l'introduction, les méthodes de Broyden résolvent le système d'équations non-linéaires suivant :

$$F(x) = 0, \text{ où } F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$$

En fait, ce sont des méthodes de type quasi-Newton, c'est-à-dire dérivées de la méthode Newton.

Méthode de Newton

La méthode de Newton a exactement le même but que les méthodes de Broyden.

Le développement de la fonction F en série de Taylor d'ordre 1 en un point $x_k \in \mathbb{R}^n$ fournit un modèle linéaire de F autour de x_k :

$$F(x) = F(x_k) + F'(x_k)(x - x_k) \tag{1}$$

En approximant F de la sorte :

$$\begin{aligned} F(x) = 0 &\Leftrightarrow F(x_k) + F'(x_k)(x - x_k) = 0 \\ &\Leftrightarrow F'(x_k)(x - x_k) = -F(x_k) \\ &\Leftrightarrow x - x_k = -(F'(x_k))^{-1}F(x_k) \\ &\Leftrightarrow x = x_k - (F'(x_k))^{-1}F(x_k) \end{aligned}$$

En posant $x_{k+1} = x$, on obtient une nouvelle approximation du zéro de F , et du même coup la récurrence suivante :

$$x_{k+1} = x_k - (F'(x_k))^{-1}F(x_k)$$

Toutefois, à chaque itération, l'algorithme doit calculer le jacobien $F'(x_k)$ de F en x_k , ce qui n'est pas forcément très facile en pratique :

D'une part, lorsque le système à résoudre atteint des dimensions très élevées, le temps de calcul du jacobien de F est très long.

D'autre part, la fonction F peut être du type "boîte noire", c'est-à-dire qu'elle n'est pas connue explicitement : elle prend simplement un point x en argument et renvoie un point $F(x)$, sans que l'on sache quelles opérations ont été effectuées entretemps.

Dans ce cas-là, le jacobien est alors tout bonnement impossible à calculer.

Le principe des méthodes de Broyden est simplement d'approximer ce jacobien de manière récurrente, pour ne pas avoir besoin de le recalculer à chaque pas.

Méthodes de Broyden

Récrivons l'équation (1) en remplaçant x par x_{k+1} et $F'(x_k)$ par B_{k+1} :

$$\begin{aligned} F(x_{k+1}) &= F(x_k) + B_{k+1}(x_{k+1} - x_k) \\ \text{i.e } F(x_{k+1}) - F(x_k) &= B_{k+1}(x_{k+1} - x_k) \end{aligned}$$

En posant $y_k = F(x_{k+1}) - F(x_k)$ et $s_k = x_{k+1} - x_k$, on obtient l'équation :

$$y_k = B_{k+1}s_k \tag{2}$$

C'est l'équation qui est à la base des deux méthodes de Broyden, nommée "équation séquante".

Dans le cas d'une fonction $F : \mathbb{R} \rightarrow \mathbb{R}$, la direction B_{k+1} est déterminée par la droite passant par les deux points $F(x_k)$ et $F(x_{k+1})$. Elle est dans ce cas uniquement déterminée.

En dimension supérieure, le principe est le même : B_{k+1} est un hyperplan passant par les points $F(x_k)$ et $F(x_{k+1})$ dans \mathbb{R}^n . Toutefois, il en existe une infinité, et le fondement des deux méthodes de Broyden réside dans le choix d'un B_{k+1} particulier.

Première méthode de Broyden

Si les points x_k et x_{k+1} sont connus, l'équation (2) ne détermine pas complètement B_k , qui possède n^2 composantes soumises à n équations.

Afin de lever cette sous-détermination sur $n^2 - n$ composantes, Broyden minimise le changement sur le modèle M , qui est l'approximation linéaire de F suivante :

$$M_k(x) = F(x_k) + B_k(x - x_k)$$

On remarque notamment que :

$$\begin{aligned} \text{a) } M_k(x_k) &= F(x_k) + B_k(x_k - x_k) \\ &= F(x_k) \\ \text{b) } M_k(x_{k+1}) &= F(x_k) + B_k(x_{k+1} - x_k) \\ &= F(x_{k+1}) \end{aligned}$$

En considérant le changement sur le modèle M , c'est-à-dire la différence $M_{k+1} - M_k$:

$$\begin{aligned} M_{k+1}(x) - M_k(x) &= [F(x_{k+1}) + B_{k+1}(x - x_{k+1})] - [F(x_k) + B_k(x - x_k)] \\ &= F(x_{k+1}) + B_{k+1}x - B_{k+1}x_{k+1} - F(x_k) - B_k(x - x_k) \\ &= F(x_{k+1}) - F(x_k) - B_{k+1}x_{k+1} + B_{k+1}x - B_k(x - x_k) \\ &= y_k - B_{k+1}x_{k+1} + B_{k+1}x_k - B_{k+1}x_k + B_{k+1}x \\ &\quad - B_k(x - x_k) \\ &= y_k - B_{k+1}(x_{k+1} - x_k) + B_{k+1}(x - x_k) - B_k(x - x_k) \\ &= y_k - B_{k+1}s_k + (B_{k+1} - B_k)(x - x_k) \\ &= (B_{k+1} - B_k)(x - x_k) \end{aligned}$$

Rappel géométrique : étant donnés deux vecteurs a et b dans \mathbb{R}^n , il est toujours possible d'en trouver un troisième c , qui est orthogonal à b , ainsi qu'un scalaire $\lambda \in \mathbb{R}$, tels que $a = \lambda b + c$.

En posant $a = x - x_k$ et $b = s_k$, on a donc que :

$$x - x_k = \lambda s_k + c, \text{ où } s_k^T c = 0$$

Ainsi :

$$\begin{aligned}
M_{k+1}(x) - M_k(x) &= (B_{k+1} - B_k)(x - x_k) \\
&= (B_{k+1} - B_k)(\lambda s_k + c) \\
&= \lambda(B_{k+1} - B_k)s_k + (B_{k+1} - B_k)c \\
&= \lambda(B_{k+1}s_k - B_k s_k) + (B_{k+1} - B_k)c \\
&= \lambda(y_k - B_k s_k) + (B_{k+1} - B_k)c
\end{aligned}$$

L'idée de Broyden est alors de minimiser la norme de cette différence. La norme choisie est celle de Frobenius, c'est-à-dire :

$$\|A\| = \left(\sum_{i=1}^n \sum_{j=1}^n (A_{ij})^2 \right)^{1/2}, \quad \forall A \in M_n(\mathbb{R})$$

Le premier terme étant déterminé, puisque dépendant uniquement du k^e pas, cela revient à rendre le second terme nul, c'est-à-dire choisir B_{k+1} de telle façon que :

$$(B_{k+1} - B_k)c = 0$$

Or, on sait que :

$$s_k^T c = 0$$

En choisissant B_{k+1} de telle manière que

$$B_{k+1} - B_k = ds_k^T, \quad \text{où } d \in \mathbb{R}^n, \quad (3)$$

la propriété voulue est bien vérifiée.

En effet, de cette façon :

$$\begin{aligned}
(B_{k+1} - B_k)c &= ds_k^T c \\
&= d \cdot 0 \\
&= 0
\end{aligned}$$

On a donc :

$$\begin{aligned}
y_k - B_k s_k &= B_{k+1} s_k - B_k s_k \\
&= (B_{k+1} - B_k) s_k \\
&= ds_k^T s_k \\
\Rightarrow d &= \frac{y_k - B_k s_k}{s_k^T s_k}
\end{aligned}$$

Or : (3) $B_{k+1} - B_k = ds_k^T$

$$\begin{aligned}\Rightarrow B_{k+1} &= B_k + ds_k^T \\ &= B_k + \frac{y_k - B_k s_k}{s_k^T s_k} s_k^T \\ &= B_k + (y_k - B_k s_k) \frac{s_k^T}{s_k^T s_k}\end{aligned}$$

C'est la relation récurrente pour B , l'approximation du jacobien.

En fait, comme cité dans [2], cette mise à jour de Broyden représente le changement minimal par rapport à B_k compte tenu du fait que $B_k s_k = y_k$, si le changement est mesuré avec la norme de Frobenius.

La preuve de ceci se trouve en [2], lemme 8.1.1 p 171.

L'algorithme de la première méthode de Broyden est alors le suivant :

Algorithme BGM (Broyden good method) :

- donné :
 - fonction $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$
 - point initial $x_1 \in \mathbb{R}^n$
 - matrice initiale $B_1 \in M_n(\mathbb{R})$
- pour k de 1 à ... :
 - résoudre $B_k s_k = -F(x_k)$ en fonction de s_k
 - $x_{k+1} = x_k + s_k$
 - $y_{k+1} = F(x_{k+1}) - F(x_k)$
 - $B_{k+1} = B_k + (y_k - B_k s_k) \frac{s_k^T}{s_k^T s_k}$

Reste à définir un critère d'arrêt, qui sera décrit dans le chapitre "Programme de tests".

Seconde méthode de Broyden

Reprenons l'équation (2) :

$$\begin{aligned}y_k &= B_{k+1} s_k \\ \Leftrightarrow B_{k+1}^{-1} y_k &= s_k\end{aligned}$$

Broyden considère alors cette fois l'approximation linéaire N suivante :

$$N_k(x) = x_k + B_k^{-1}(F(x) - F(x_k))$$

On remarque que :

$$\begin{aligned} \text{a) } N_k(x_k) &= x_k + B_k^{-1}(F(x_k) - F(x_k)) \\ &= x_k \\ \text{b) } N_k(x_{k+1}) &= x_k + B_k^{-1}(F(x_{k+1}) - F(x_k)) \\ &= x_{k+1} \end{aligned}$$

Le calcul de la différence entre deux modèles successifs $N_{k+1} - N_k$ est tout à fait similaire à celui de $M_{k+1} - M_k$:

$$\begin{aligned} N_{k+1}(x) - N_k(x) &= \dots \\ &= (B_{k+1}^{-1} - B_k^{-1})(F(x) - F(x_k)) \end{aligned}$$

Comme pour la première méthode, on peut trouver $\lambda \in \mathbb{R}$ et $c \in \mathbb{R}^n$ avec $y_k^T c = 0$ tels que :

$$\begin{aligned} F(x) - F(x_k) &= \lambda(F(x_{k+1}) - F(x_k)) + c \\ &= \lambda y_k + c \end{aligned}$$

Ainsi :

$$\begin{aligned} N_{k+1}(x) - N_k(x) &= \dots \\ &= \lambda(s_k - B_k^{-1}y_k) + (B_{k+1}^{-1} - B_k^{-1})c \end{aligned}$$

A nouveau, pour minimiser la norme de cette différence, il faut rendre le second terme nul, puisque le premier est déjà déterminé au k^e pas. Le raisonnement est alors exactement le même que pour la première méthode.

Finalement, la relation pour B_{k+1}^{-1} est la suivante :

$$B_{k+1}^{-1} = B_k^{-1} + (s_k - B_k^{-1}y_k) \frac{y_k^T}{y_k^T y_k}$$

Le cheminement est donc exactement le même que pour la première méthode, sauf que les modèles M et N diffèrent : le premier s'inspire de la relation $y_k = B_{k+1}s_k$ et le second de $s_k = B_{k+1}^{-1}y_k$.

Voici maintenant l'algorithme de la seconde méthode :

Algorithme BBM (Broyden bad method) :

– donné :

- fonction $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$
- point initial $x_1 \in \mathbb{R}^n$
- matrice initiale $B_1^{-1} \in M_n(\mathbb{R})$
- pour k de 1 à ... :
 - $s_k = -B_k^{-1}F(x_k)$
 - $x_{k+1} = x_k + s_k$
 - $y_{k+1} = F(x_{k+1}) - F(x_k)$
 - $B_{k+1}^{-1} = B_k^{-1} + (s_k - B_k^{-1}y_k) \frac{y_k^T}{y_k^T y_k}$

Tout ce raisonnement est tiré de [2].

Il existe une variante de chacun des deux algorithmes présentés précédemment.

On peut, dans l'algorithme BGM, calculer directement l'inverse de la matrice B_{k+1} pour ne pas devoir résoudre de système linéaire.

Inversément, une variante de l'algorithme BBM consiste à calculer B_{k+1} puis résoudre un système linéaire.

Voici comment les relations récurrentes sont obtenues :

$$\text{BGM : } B_{k+1} = B_k + (y_k - B_k s_k) \frac{s_k^T}{s_k^T s_k}$$

On applique alors le lemme de Sherman-Morrison-Woodbury (voir [2] p 188) où : $A = B_k$, $u = y_k - B_k s_k$, $v = s_k$:

$$\begin{aligned} B_{k+1}^{-1} &= B_k^{-1} - \frac{1}{s_k^T s_k + s_k^T B_k^{-1} (y_k - B_k s_k)} B_k^{-1} (y_k - B_k s_k) s_k^T B_k^{-1} \\ &= B_k^{-1} - \frac{1}{s_k^T s_k + s_k^T B_k^{-1} y_k - s_k^T B_k^{-1} B_k s_k} (B_k^{-1} y_k - B_k^{-1} B_k s_k) s_k^T B_k^{-1} \\ &= B_k^{-1} - \frac{1}{s_k^T s_k + s_k^T B_k^{-1} y_k - s_k^T s_k} (B_k^{-1} y_k - s_k) s_k^T B_k^{-1} \\ &= B_k^{-1} + (s_k - B_k^{-1} y_k) \frac{s_k^T B_k^{-1}}{s_k^T B_k^{-1} y_k} \end{aligned}$$

On obtient donc l'algorithme suivant :

Algorithme BGM_bis (variante de BGM) :

- donné :
 - fonction $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$
 - point initial $x_1 \in \mathbb{R}^n$

- matrice initiale $B_1^{-1} \in M_n(\mathbb{R})$
- pour k de 1 à ... :
 - $s_k = -B_k^{-1}F(x_k)$
 - $x_{k+1} = x_k + s_k$
 - $y_{k+1} = F(x_{k+1}) - F(x_k)$
 - $B_{k+1}^{-1} = B_k^{-1} + (s_k - B_k^{-1}y_k) \frac{s_k^T B_k^{-1}}{s_k^T B_k^{-1} y_k}$

Appliquant, comme auparavant, le lemme de Sherman-Morrison-Woodbury à la relation récurrente de l'algorithme BBM, on trouve l'expression suivante :

$$B_{k+1} = B_k + (y_k - B_k s_k) \frac{y_k^T B_k}{y_k^T B_k s_k}$$

Ainsi :

Algorithme BBM_bis (variante de BBM) :

- donné :
 - fonction $F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$
 - point initial $x_1 \in \mathbb{R}^n$
 - matrice initiale $B_1 \in M_n(\mathbb{R})$
- pour k de 1 à ... :
 - résoudre $B_k s_k = -F(x_k)$ en fonction de s_k
 - $x_{k+1} = x_k + s_k$
 - $y_{k+1} = F(x_{k+1}) - F(x_k)$
 - $B_{k+1} = B_k + (y_k - B_k s_k) \frac{y_k^T B_k}{y_k^T B_k s_k}$

Résumé

Voici donc les quatre relations récurrentes pour les matrices B_{k+1} ou B_{k+1}^{-1} :

BGM $B_{k+1} = B_k + (y_k - B_k s_k) \frac{s_k^T}{s_k^T s_k}$	BBM_bis $B_{k+1} = B_k + (y_k - B_k s_k) \frac{y_k^T B_k}{y_k^T B_k s_k}$
BGM_bis $B_{k+1}^{-1} = B_k^{-1} + (s_k - B_k^{-1} y_k) \frac{s_k^T B_k^{-1}}{s_k^T B_k^{-1} y_k}$	BBM $B_{k+1}^{-1} = B_k^{-1} + (s_k - B_k^{-1} y_k) \frac{y_k^T}{y_k^T y_k}$

Les deux mises à jour de B_{k+1} sont similaires, de même que celles de B_{k+1}^{-1} . La seule différence réside dans le terme fractionnel de droite.

Un problème peut toutefois apparaître dans l'algorithme BBM ; en effet, lorsqu'il se rapproche du \tilde{x} tel que $F(\tilde{x}) = 0$, $y^T y = \|y\|^2$ devient très petit, et de ce fait B_{k+1}^{-1} peut prendre des valeurs aberrantes, puisque sa mise à jour effectue une division par un nombre qui tend vers 0.

Par contre, ce problème n'apparaît pas dans l'algorithme BGM.

C'est peut-être une première explication du fait que la seconde méthode est usuellement qualifiée de "mauvaise".

Origine des termes “bonne” et “mauvaise” méthodes

Comme cité en introduction, une des deux méthodes est systématiquement nommée la “bonne” méthode de Broyden et l’autre la “mauvaise”. L’origine de ces termes reste toutefois très mystérieuse.

Dans son premier article traitant le sujet (voir [1]) et datant de 1965, Broyden présente les deux méthodes en les nommant “method 1” et “method 2”. Voici ce qu’il dit au sujet de la seconde méthode :

This method appears in practice to be unsatisfactory, it will be discussed no further at this stage.

[...]

While the program was being developed, it became increasingly apparent that method 2 was quite useless.

In every case I tried it rapidly reached a state where successive step vectors s_i and s_{i+1} were identical and further progress thus became impossible, and for this reason it is omitted from the following case histories.

Broyden n’effectue donc aucun test sur cette méthode ; c’est probablement de là que sont issus les termes. Je n’ai malheureusement pas réussi à retrouver les articles suivants de Broyden.

La question de cette origine a été posée à J.M. Martinez par mail, et voici ce qu’il a répondu :

In the original paper by Broyden both methods were presented without the denomination “good” and “bad”. They have been known also as “first” and “second” methods of Broyden respectively.

Numerical experiments in the first years after their introduction suggested

that the "good" method was far better than the "bad" method. After that, many authors began to call them "good" and "bad".

In fact, I do not know exactly who was the first to publish those names, although I am sure that they were called in that way informally in meetings and congresses.

Conjectures de J.M. Martinez

J.M. Martinez est un mathématicien brésilien qui a publié plusieurs articles concernant les méthodes de type quasi-Newton et celles de Broyden, tels que [4], [5], [6] et [7].

Dans un article datant de 2000 (voir [3]), il pose deux conjectures, censées déterminer dans quels cas la première méthode fonctionne mieux que la seconde.

Première conjecture

J.M. Martinez considère le terme $B_{k+1}^{-1}y_{k-1} - s_{k-1}$ dans les deux méthodes :

$$\begin{aligned}
 \text{BGM_bis} & : B_{k+1}^{-1}y_{k-1} - s_{k-1} = (B_k^{-1} + (s_k - B_k^{-1}y_k) \frac{s_k^T B_k^{-1}}{s_k^T B_k^{-1}y_k})y_{k-1} - s_{k-1} \\
 & = B_k^{-1}y_{k-1} + (s_k - B_k^{-1}y_k) \frac{s_k^T B_k^{-1}y_{k-1}}{s_k^T B_k^{-1}y_k} - s_{k-1} \\
 & = (s_k - B_k^{-1}y_k) \frac{s_k^T s_{k-1}}{s_k^T B_k^{-1}y_k} \\
 \text{BBM} & : B_{k+1}^{-1}y_{k-1} - s_{k-1} = (B_k^{-1} + (s_k - B_k^{-1}y_k) \frac{y_k^T}{y_k^T y_k})y_{k-1} - s_{k-1} \\
 & = B_k^{-1}y_{k-1} + (s_k - B_k^{-1}y_k) \frac{y_k^T}{y_k^T y_k} y_{k-1} - s_{k-1} \\
 & = (s_k - B_k^{-1}y_k) \frac{y_k^T y_{k-1}}{y_k^T y_k}
 \end{aligned}$$

Il remarque alors que $B_{k+1}^{-1}y_{k-1} - s_{k-1}$, qu'il appelle "erreur sécante", est dans les deux cas un multiple de $s_k - B_k^{-1}y_k$.

Selon lui, plus l'erreur séquante est faible, mieux les méthodes de Broyden fonctionnent. Il postule donc que BGM sera plus efficace que BBM si :

$$\left| \frac{s_k^T s_{k-1}}{s_k^T B_k^{-1}y_k} \right| < \left| \frac{y_k^T y_{k-1}}{y_k^T y_k} \right| \quad (4)$$

Seconde conjecture de Martinez

J.M Martinez obtient une deuxième conjecture en faisant la même comparaison sur le terme $B_{k+1}s_{k-1} - y_{k-1}$.

$$\begin{aligned}
 \text{BGM} \quad : \quad B_{k+1}s_{k-1} - y_{k-1} &= (B_k + (y_k - B_k s_k) \frac{s_k^T}{s_k^T s_k}) s_{k-1} - y_{k-1} \\
 &= B_k s_{k-1} + (y_k - B_k s_k) \frac{s_k^T}{s_k^T s_k} s_{k-1} - y_{k-1} \\
 &= (y_k - B_k s_k) \frac{s_k^T s_{k-1}}{s_k^T s_k} \\
 \text{BBM_bis} \quad : \quad B_{k+1}s_{k-1} - y_{k-1} &= (B_k + (y_k - B_k s_k) \frac{y_k^T B_k}{y_k^T B_k s_k}) s_{k-1} - y_{k-1} \\
 &= B_k s_{k-1} + (y_k - B_k s_k) \frac{y_k^T B_k s_{k-1}}{y_k^T B_k s_k} - y_{k-1} \\
 &= (y_k - B_k s_k) \frac{y_k^T y_{k-1}}{y_k^T B_k s_k}
 \end{aligned}$$

A nouveau, $B_{k+1}s_{k-1} - y_{k-1}$ est dans les deux cas le multiple d'un même terme, en l'occurrence $y_k - B_k s_k$. Il postule donc que BGM fonctionnera mieux que BBM si :

$$\left| \frac{s_k^T s_{k-1}}{s_k^T s_k} \right| < \left| \frac{y_k^T y_{k-1}}{y_k^T B_k s_k} \right| \quad (5)$$

Dans [3], J.M. Martinez créé un algorithme qui choisit à chaque pas quel algorithme il utilise en fonction de (4) et (5). Cet algorithme s'est, selon ses dires, révélé plus efficace que BGM et BBM.

Pour ma part, ces conjectures ne me semblent pas très significatives. Il n'est en effet pas évident que la taille de l'erreur séquante est seule responsable du bon fonctionnement de l'une ou l'autre méthode.

Une analyse géométrique de ces deux conjectures a été réalisée, mais elle n'a pas été concluante, c'est-à-dire que les termes des conjectures ne représentent rien de vraiment particulier.

Programme de tests

Pour tester les deux méthodes de Broyden, j'ai implémenté un programme sur Matlab dont voici le descriptif :

Le programme principal, nommé *main*, spécifie :

la fonction F à tester :

La liste des fonctions testées se trouve en annexe 1.

le point initial x_1 :

La plupart du temps, le point initial est donné dans la définition de la fonction. Si ce n'est pas le cas, il faut essayer de le fixer "assez proche" du zéro de la fonction, pour que l'algorithme ne diverge pas.

la matrice initiale B_1 :

La matrice initiale a été choisie comme étant l'identité.

le premier critère d'arrêt :

Le critère d'arrêt choisi a été la norme de F . Cela semble en effet légitime, car le but est de trouver x tel que $F(x) = 0$, et de plus $F(x) = 0 \iff \|F(x)\| = 0$. L'algorithme s'arrête donc lorsque la norme de $F(x)$ est inférieure à 10^{-6} (comme Broyden dans [1]).

le second critère d'arrêt :

Pour prévenir les cas de divergence, l'algorithme stoppe après 500 itérations.

Avec ces données, *main* lance les algorithmes *bgm* et *bbm*, qui fonctionnent comme décrit au chapitre "Méthodes de Broyden" avec les deux critères d'arrêt définis précédemment, et qui calculent de plus les termes des conjectures

de Martinez.

main lance aussi les algorithmes *bgm_bis* et *bbm_bis*, qui fonctionnent comme *bgm* et *bbm*, mais utilisent respectivement la mise à jour de B_k^{-1} et B_k .

Les différents résultats que peut retourner le programme *main* sont les suivants :

- a) Pour les algorithmes *bgm* et *bbm*, la suite des x_i , $F(x_i)$ et $\|F(x_i)\|$.
- b) Les valeurs successives des termes des deux conjectures de Martinez.
- c) Une synthèse comprenant :
 - le nom de la fonction
 - le point initial
 - la matrice initiale
 - le nombre d'itérations nécessaires à *bgm* et à *bbm*
 - les zéros trouvés dans les deux cas
 - les nombres d'itérations et les zéros déterminés par *bgm_bis* et *bbm_bis*, dans le cas où ces derniers ont mieux fonctionné que *bgm* et *bbm* respectivement.
- d) Dans le cas d'une fonction de dimension 2, les graphes des x_i et $F(x_i)$ successifs dans le plan pour *bgm* et *bbm*.

Fonctions linéaires

Etudions pour commencer les deux méthodes sur des fonctions linéaires.

Voici les relations récurrentes pour les matrices :

$$\text{BGM} : B_{k+1} = B_k + (y_k - B_k s_k) \frac{s_k^T}{s_k^T s_k} \quad (6)$$

$$\text{BBM} : B_{k+1}^{-1} = B_k^{-1} + (s_k - B_k^{-1} y_k) \frac{y_k^T}{y_k^T y_k} \quad (7)$$

On remarque que si on échange les termes y_k et s_k dans (6), on obtient (7), à ceci près que l'on a des B^{-1} à la place des B .

La première idée a donc été que BBM est en quelque sorte le “symétrique” de BGM, c’est-à-dire que BGM “travaille” sur les x_k et BBM sur les $F(x_k)$.

Si c’est le cas, la résolution du système $F^{-1}(x) = 0$ devrait être similaire à celle de $F(x) = 0$. C’est ceci qui est développé par la suite :

Soit donc $F(x) = Ax$ une fonction linéaire.

Prenons un point initial x_1 et une matrice initiale B_1 .

Pour la fonction inverse $F^{-1}(x) = A^{-1}x$, il semble logique de prendre comme point initial $F(x_1) = Ax_1$.

Quant à la matrice initiale, de manière similaire au point initial, il semble raisonnable de prendre $A \cdot B_1$, c’est-à-dire l’équivalent de B_1 du côté de l’image de F .

J’ai testé cette théorie en dimension 2, en explicitant les calculs pendant une ou deux étapes, avec les paramètres suivants :

$$F(x) = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} x \quad , \quad x_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad , \quad B_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Les calculs pour les deux algorithmes se trouvent en annexe 2.

Toutefois, en faisant les mêmes calculs pour F^{-1} , (cf annexe 3), on ne constate hélas aucune similarité entre les deux développements.

Les mêmes calculs ont été effectués en prenant les mêmes point ou matrice initiaux pour F et F^{-1} , mais cela n'a rien amené de plus.

Il semble donc que l'hypothèse de "symétrie" entre BGM et BBM est erronée.

En étudiant plus attentivement les relations récurrentes pour B_{k+1}^{-1} dans le cas $F(x) = Ax$:

$$F(x_k) = Ax_k \text{ et } F(x_{k+1}) = Ax_{k+1}$$

$$\begin{aligned} \Rightarrow y_k &= F(x_{k+1}) - F(x_k) \\ &= Ax_{k+1} - Ax_k \\ &= A(x_{k+1} - x_k) \\ &= As_k \end{aligned}$$

$$\begin{aligned} \text{BGM.bis : } B_{k+1}^{-1} &= B_k^{-1} + (s_k - B_k^{-1}y_k) \frac{s_k^T B_k^{-1}}{s_k^T B_k^{-1} y_k} \\ &= B_k^{-1} + (s_k - B_k^{-1}As_k) \frac{s_k^T B_k^{-1}}{s_k^T B_k^{-1} As_k} \end{aligned}$$

$$\begin{aligned} \text{BBM : } B_{k+1}^{-1} &= B_k^{-1} + (s_k - B_k^{-1}y_k) \frac{y_k}{y_k^T y_k} \\ &= B_k^{-1} + (s_k - B_k^{-1}As_k) \frac{(As_k)^T}{(As_k)^T As_k} \\ &= B_k^{-1} + (s_k - B_k^{-1}As_k) \frac{s_k^T A^T}{s_k^T A^T As_k} \end{aligned}$$

Les deux mises à jour des matrices pour B_{k+1}^{-1} sont donc les mêmes si $A^T = B_k^{-1}$. On constate de plus la même propriété si l'on compare les mises à jour de la matrice B_k .

En vérifiant ceci avec les calculs (voir en annexe 4), on constate en effet que les deux algorithmes sont dans ce cas exactement les mêmes.

En effectuant des tests sur d'autres fonctions linéaires $F(x) = Ax$, je suis arrivé aux résultats suivants :

Tout d'abord, si l'on prend pour matrice initiale A^{-T} , les deux algorithmes parviennent systématiquement au zéro de F en un même nombre d'itérations, quelquesoit le point initial choisi.

Ensuite, si la matrice initiale est l'identité, les deux algorithmes nécessitent, comme précédemment, le même nombre d'itérations dans la plupart des cas. Toutefois, en testant les méthodes sur de "gros" exemples, tels que

$$F(x) = \begin{pmatrix} 1111 & -2222 \\ 1111 & 3333 \end{pmatrix} x,$$

il s'est avéré que BBM nécessitait plus d'itérations que BGM, et l'écart s'accroît de plus en plus au fur et à mesure que les composantes de la matrices grandissent.

Afin d'étudier les conjectures de Martinez, qui ont notamment été testées dans le cas linéaire, prenons la même fonction que précédemment :

$$F(x) = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} x \quad , \quad x_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad , \quad B_1 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Voici les termes des conjectures à chaque itération :

	Conjecture 1		Conjecture 2	
k	$\left \frac{s_k^T s_{k-1}}{s_k^T B_k^{-1} y_k} \right $	$\left \frac{y_k^T y_{k-1}}{y_k^T y_k} \right $	$\left \frac{s_k^T s_{k-1}}{s_k^T s_k} \right $	$\left \frac{y_k^T y_{k-1}}{y_k^T B_k s_k} \right $
1	0	0	0	0
2	1.0275	1.9903	0.9932	1.2772
3	0.0988	0.6881	0.1639	1.4067
4	2.5170	1.9575	2.5170	1.9575

Aucune de ces deux conjectures n'est en fait très significative, puisque pour chacune d'elles, le terme de gauche est tantôt supérieur, tantôt inférieur au terme de droite.

En fait, le principe des deux conjectures est d'exprimer à chaque pas quelle méthode fonctionnera le mieux à l'itération suivante, mais pas laquelle sera la plus rapide sur l'ensemble de la résolution.

Ces conjectures ne seront donc pas débattues plus longuement dans ce rapport.

Tests sur d'autres fonctions

Fonctions de Broyden :

Voici les résultats des tests effectués sur les fonctions données par Broyden dans [1] :

fonction	nG	nB
broyden65_cas5	67	25
broyden65_cas6	> 500	> 500
broyden65_cas7	> 500	> 500
broyden65_cas8	> 500	> 500
broyden65_cas9	14	24
broyden65_cas10	58	> 500

nG = nombre d'itérations nécessaires à BGM ou BGM_bis

nB = nombre d'itérations nécessaires à BBM ou BBM_bis

Fonctions de Spedicato :

La dimension des fonctions issues de [8] étant libre, voici une analyse de quelques-unes d'entre elles prises au hasard, en faisant varier la dimension :

fonction	dimension	nG	nB
spedicato_1	3	49	> 500
	4 et plus	> 500	> 500
spedicato_4	2	11	12
	10	11	12
	20	12	12
	100	13	14
spedicato_12	5	6	6
	10	6	6
	100	6	6
spedicato_17	3	96	27
	4 et plus	> 500	> 500
spedicato_20	2	101	14
	3 et plus	> 500	> 500

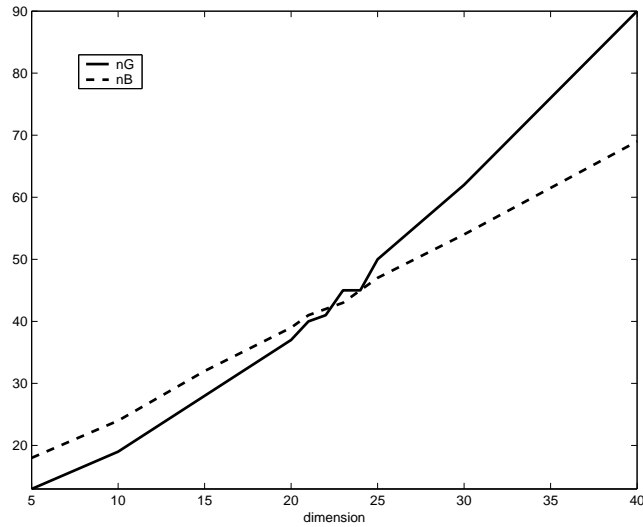
La fonction spedicato_17 ne fournit pas les mêmes résultats dans [8]. A ce titre, voici la réponse de son auteur lui-même :

The algorithms were programmed by dr. Huang, who was a very bad programmer, [...] so it must be a programming mistake that escaped my attention.

La fonction spedicato_26 est intéressante, car pour des dimensions faibles, BGM fonctionne mieux que BBM, mais cette tendance s'inverse lorsque la dimension augmente.

Le tableau suivant représente le nombre d'itérations nécessaires à chaque algorithme en fonction de la dimension du problème.

n	5	10	15	20	21	22	23	24	25	30	40
BGM	13	19	28	37	40	41	45	45	50	62	90
BBM	18	24	32	39	41	42	43	45	47	54	69



Fonctions diverses :

Voici tout d'abord deux fonctions créées par Martinez, dont le point initial est variable.

fonction	dimension	point initial	nG	nB
martinez59Pb9	6	$(0 \dots 0)^T$	> 501	> 501
	6	$(10 \dots 10)^T$	88	35
	6	$(-50 \dots -50)^T$	68	30
	6	$(-100 \dots -100)^T$	75	31
martinez59Pb13	3	$(1 \ 1 \ 1)^T$	9	11
	6	$(1 \ \dots \ 1)^T$	19	30
	10	$(1 \ \dots \ 1)^T$	25	53
	3	$(0 \ 0 \ 0)^T$	9	10
	6	$(0 \ \dots \ 0)^T$	18	29
	10	$(0 \ \dots \ 0)^T$	23	51

Ensuite, voici deux fonctions dont le point initial est variable :

fonction	point initial	nG	nB
roux_1	$(0 \ 4)^T$	30	14
	$(1 \ 1)^T$	19	9
	$(14 \ 14)^T$	16	12
	$(-3 \ -3)^T$	11	9
	$(0.5 \ 0.5)^T$	8	8
	$(100 \ 100)^T$	8	9
	$(-1 \ -1)^T$	10	11
	$(-7 \ -89)^T$	8	10
roux_4	$(1 \ 2 \ 1)^T$	12	> 500
	$(4 \ 0 \ -2)^T$	15	> 500
	$(1 \ -1 \ 1)^T$	16	> 500
	$(100 \ 99 \ 98)^T$	49	38
	$(890 \ 132 \ 470)^T$	80	20
	$(89'000 \ 13'200 \ 47'000)^T$	27	> 500

Les fonctions suivantes n'ont ni point initial, ni dimension fixée :

fonction	dimension	point initial	nG	nB
roux_2	5	$(1 \ 2 \ 1 \ 2 \ 1)^T$	32	25
	10	$(1 \ 2 \ \dots \ 2 \ 1)^T$	35	31
	5	$(-1 \ 1 \ -1 \ 1 \ -1)^T$	21	25
	10	$(-1 \ 1 \ \dots \ 1 \ -1)^T$	21	25
	5	$(0 \ 8 \ 0 \ 8 \ 0)^T$	24	24
	10	$(0 \ 8 \ \dots \ 0 \ 8)^T$	25	25
roux_3	2	$(-1 \ 3)^T$	4	4
	3	$(-4 \ 30 \ -2)^T$	7	4
	4	$(-4 \ 3 \ -2 \ 1)^T$	57	4
	5	$(-1 \ 3 \ -9 \ 8 \ -7)^T$	8	4
	10	$(-1 \ \dots \ -1)^T$	3	3

Voici pour finir diverses autres fonctions :

fonction	dimension	nG	nB
badScale	4	118	150
	6	286	459
dennis	2	13	14
faresBurden413	3	17	15
HVF3	2	120	> 500
nocedal	2	46	43

Analyse des résultats

Les conclusions suivantes peuvent être tirées des résultats obtenus.

Premièrement, Broyden a dû commettre une erreur dans ses tests, car la fonction `broyden65_cas5`, issue de son propre ouvrage de 1965 ([1]), est plus rapidement résolue par BBM que par BGM.

C'est toutefois la seule des fonctions qu'il explicite qui est dans ce cas.

En ce qui concerne les fonctions linéaires, les deux méthodes utilisent en règle générale un même nombre d'itérations.

Toutefois, la seconde méthode présente le défaut d'effectuer une division par $\|y_k\|^2$ lors de la mise à jour de la matrice, ce qui pose de gros problèmes d'arrondis sur certaines fonctions.

La fonction `spedicato_26` est très révélatrice ; elle prouve en effet qu'une même fonction peut mieux convenir à une méthode en faible dimension, et moins bien lorsque cette dimension augmente.

La même remarque peut être faite au sujet du point initial : les fonctions `roux_1` et `roux_4` en sont à ce titre de très bons exemples.

Enfin, les tests effectués sur toutes les autres fonctions montrent que la première méthode n'est de loin pas plus efficace que la seconde.

Conclusion

Au vu de ce qui précède, il me semble indéniable que la seule conclusion qui peut être tirée est qu'il n'y a pas de "bonne" et de "mauvaise" méthode de Broyden : toutes deux fonctionnent très bien.

Certaines fonctions, comme notamment les fonctions linéaires, sont résolues plus rapidement par la première méthode, alors que pour d'autres c'est la seconde méthode qui est la plus adaptée.

Ce qui a aussi été vu est que les deux méthodes sont très sensibles au choix du point initial choisi, ainsi qu'à la dimension du problème. Il en est d'ailleurs certainement de même avec le choix de la matrice initiale.

En fait, les termes "bonne" et "mauvaise" méthodes sont initialement dus à Broyden, qui a lui-même affirmé que la seconde ne fonctionnait pas bien. Par la suite, les tests effectués étaient très certainement involontairement favorables à la première méthode, ce qui a renforcé son appellation de "bonne" méthode.

Je terminerai en remerciant Michela Spada et Frank Crittin pour leur excellent coaching durant toute la durée du semestre.

Annexes

Annexe 1 : liste des fonctions :

broyden65_cas5 à broyden65_cas8 :

- origine : [1] p 587
- dimension : n
- point initial : $(-1 \dots -1)^T$

$$F_i(x) = \begin{cases} -(3 + \alpha x_1)x_1 + 2x_2 - \beta & \text{si } i = 1 \\ x_{i-1} - (3 + \alpha x_i)x_i + 2x_{i+1} - \beta & \text{si } i = 2, \dots, n-1 \\ x_{n-1} - (3 + \alpha x_n)x_n - \beta & \text{si } i = n \end{cases}$$

- cas 5 : $\alpha = -0.1$, $\beta = 1$, $n = 5$
- cas 6 : $\alpha = -0.5$, $\beta = 1$, $n = 5$
- cas 7 : $\alpha = -0.5$, $\beta = 1$, $n = 10$
- cas 8 : $\alpha = -0.5$, $\beta = 1$, $n = 20$

broyden65_cas9 :

- origine : [1] p 587
- dimension : 2
- point initial : $(-1.2 \ 1)^T$

$$F_i(x) = \begin{cases} 10(x_2 - x_1^2) & \text{si } i = 1 \\ 1 - x_1 & \text{si } i = 2 \end{cases}$$

broyden65_cas10 :

- origine : [1] p 587
- dimension : 2
- point initial : $(15 \quad -2)^T$

$$F_i(x) = \begin{cases} -13 + x_1 + ((-x_2 + 5)x_2 - 2)x_2 & \text{si } i = 1 \\ -29 + x_1 + ((x_2 + 1)x_2 - 14)x_2 & \text{si } i = 2 \end{cases}$$

spedicato_1 :

- origine : [8], fonction 1
- dimension : n
- point initial : $(-1.2 \dots -1.2 \quad -1)^T$

$$F_i(x) = \begin{cases} 1 - x_1 & \text{si } i = 1 \\ 10(i-1)(x_i - x_{i-1})^2 & \text{si } i = 2, \dots, n \end{cases}$$

spedicato_4 :

- origine : [8], fonction 4
- dimension : n (pair)
- point initial : $(-1.2 \quad 1 \dots -1.2 \quad 1)^T$

$$F_i(x) = \begin{cases} 1 - x_i & \text{si } i \text{ impair} \\ 10(x_i - x_{i-1}^2) & \text{si } i \text{ pair} \end{cases}$$

spedicato_12 :

- origine : [8], fonction 12
- dimension : n
- point initial : $(0.5 \dots 0.5)^T$

$$F_i(x) = \begin{cases} x_1 & \text{si } i = 1 \\ \cos x_{i-1} + x_i - 1 & \text{si } i = 2, \dots, n \end{cases}$$

spedicato_17 :

- origine : [8], fonction 17
- dimension : n
- point initial : $(10 \dots 10)^T$

$$F_i(x) = \begin{cases} 3x_1 + (x_2 - 2x_1) + \frac{x_2^2}{4} & \text{si } i = 1 \\ 3x_i + (x_{i+1} - 2x_i + x_{i-1}) + \frac{1}{4}(x_{i+1} - x_{i-1})^2 & \text{si } i = 2, \dots, n-1 \\ 3x_n + (20 - 2x_n + x_{n-1}) + \frac{1}{4}(20 - x_{n-1})^2 & \text{si } i = n \end{cases}$$

spedicato_20 :

- origine : [8], fonction 20
- dimension : n
- point initial : $(\frac{1}{n} \dots \frac{1}{n})^T$

$$F_i(x) = n - \sum_{j=1}^n \cos x_j + i(1 - \cos x_i) - \sin x_i, \text{ pour } i = 1, \dots, n$$

spedicato_26 :

- origine : [8], fonction 26
- dimension : n
- point initial : $(1 \dots 1)^T$

$$F_i(x) = \begin{cases} 2x_1 - x_2 + 10h^2 \sin(10hx_1) & \text{si } i = 1 \\ -xi - 1 + 2x_i - xi + 1 + 10h^2 \sin(10hx_i) & \text{si } i = 2, \dots, n-1 \\ -xn - 1 + 2x_n - 1 + 10h^2 \sin(10hx_n) & \text{si } i = n \end{cases}$$

$$\text{où } h = \frac{1}{n+1}$$

martinez59Pb9 :

- origine : article 59 de Martinez, problème 9
- dimension : 6
- point initial : non spécifié

$$F_i(x) = \sum_{j \neq i} \cot(b_i \cdot x_j), \text{ pour } i = 1, \dots, 6$$

$$\text{où } b = (0.02249 \quad 0.02166 \quad 0.02083 \quad 0.02 \quad 0.01918 \quad 0.01835)^T$$

martinez59Pb13 :

- origine : article 59 de Martinez, problème 13
- dimension : n
- point initial : non spécifié

$$F_i(x) = \begin{cases} (3 - 0.1x_1)x_1 + 1 - 2x_2 + x_1 & \text{si } i = 1 \\ (3 - 0.1x_i)x_i + 1 - x_{i-1} - 2x_{i+1} + x_i & \text{si } i = 2, \dots, n - 1 \\ (3 - 0.1x_n)x_n + 1 - 2x_{n-1} + x_n & \text{si } i = n \end{cases}$$

roux_1 :

- origine : Patrick Roux
- dimension : 2
- point initial : libre

$$F_i(x) = \begin{cases} \sin(x_1 + x_2) & \text{si } i = 1 \\ \cos(x_1 - x_2) & \text{si } i = 2 \end{cases}$$

roux_2 :

- origine : Patrick Roux
- dimension : n
- point initial : libre

$$F_i(x) = \left(\sum_{j=1}^n x_j - x_i \right) \cdot x_i$$

roux_3 :

- origine : Patrick Roux
- dimension : n
- point initial : libre

$$F(x) = |x|$$

roux_4 :

- origine : Patrick Roux
- dimension : 3
- point initial : libre

$$F_i(x) = x_1^i + x_2^i + x_3^i, \text{ pour } i = 1, 2, 3$$

badScale :

- origine : Frank Crittin
- dimension : n (pair)
- point initial : $(2 \dots 2)^T$

$$F_i(x) = \begin{cases} 100 \cos x_i^2 - x_i & \text{si } i \text{ impair} \\ \cos x_i \cdot \log(x_i^2 + 1) - x_i & \text{si } i \text{ pair} \end{cases}$$

dennis :

- origine : [2] p 149
- dimension : 2
- point initial : $(2 \quad \frac{1}{2})^T$

$$F_i(x) = \begin{cases} x_1^2 + x_2^2 - 2 & \text{si } i = 1 \\ e^{x_1-1} + x_2^3 - 2 & \text{si } i = 2 \end{cases}$$

faresBurden413 :

- origine : Faires & Burden p 413
- dimension : 3
- point initial : $(0 \quad 0 \quad 0)^T$

$$F_i(x) = \begin{cases} 3x_1 - \cos(x_2 \cdot x_3) - x_1 & \text{si } i = 1 \\ x_1^2 - 81(x_2 + 0.1) + \sin x_3 + 1.06 - x_2 & \text{si } i = 2 \\ e^{-x_1 \cdot x_2} + 20x_3 + \frac{10\pi-3}{3} - \frac{1}{2} - x_3 & \text{si } i = 3 \end{cases}$$

HVF3 :

- origine : [2], p 362
- dimension : 3
- point initial : $(-1 \ 0 \ 0)^T$

$$F_i(x) = \begin{cases} 10(x_3 - 10\Omega) & \text{si } i = 1 \\ 10(\sqrt{x_1^2 + x_2^2} - 1) & \text{si } i = 2 \\ x_3 & \text{si } i = 3 \end{cases}$$

$$\text{où } \Omega = \begin{cases} \frac{\pi}{2} \arctan \frac{x_2}{x_1} & \text{si } x_1 \geq 0 \\ \frac{\pi}{2} \arctan \frac{x_2}{x_1} + \frac{1}{2} & \text{si } x_1 < 0 \end{cases}$$

nocedal :

- origine : Nocedal & Wright p 288
- dimension : 2
- point initial : $(-0.5 \ 1.4)^T$

$$F_i(x) = \begin{cases} (x_1 + 3) \cdot (x_2^3 - 7) + 18 & \text{si } i = 1 \\ \sin(x_2 \cdot e^{x_1} - 1) & \text{si } i = 2 \end{cases}$$

Annexe 2 : premier calcul matriciel :

$$\begin{aligned}F(x) &= \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} x \\x_1 &= \begin{pmatrix} 1 \\ 1 \end{pmatrix} \\B_1 &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}\end{aligned}$$

Algorithme BGM :

$$B_1 s_1 = -F(x_1) : \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} s_1^1 \\ s_1^2 \end{pmatrix} = - \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\Rightarrow s_1 = \begin{pmatrix} -1 \\ 4 \end{pmatrix}$$

$$x_2 = x_1 + s_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} -1 \\ -4 \end{pmatrix} = \begin{pmatrix} 2 \\ -3 \end{pmatrix}$$

$$\Rightarrow F(x_2) = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 2 \\ -3 \end{pmatrix} = \begin{pmatrix} 8 \\ -7 \end{pmatrix}$$

$$y_1 = F(x_2) - F(x_1) = \begin{pmatrix} 8 \\ -7 \end{pmatrix} - \begin{pmatrix} -1 \\ 4 \end{pmatrix} = \begin{pmatrix} 9 \\ -11 \end{pmatrix}$$

$$B_2 = B_1 + (y_1 - B_1 s_1) \frac{s_1^T}{s_1^T s_1} = \dots = \frac{1}{17} \begin{pmatrix} 25 & -32 \\ -7 & 45 \end{pmatrix}$$

$$B_2 s_2 = -F(x_2) : \frac{1}{17} \begin{pmatrix} 25 & -32 \\ -7 & 28 \end{pmatrix} \begin{pmatrix} s_2^1 \\ s_2^2 \end{pmatrix} = - \begin{pmatrix} 8 \\ -7 \end{pmatrix}$$

$$\Rightarrow s_2 = \frac{1}{53} \begin{pmatrix} -136 \\ 119 \end{pmatrix}$$

$$x_3 = x_2 + s_2 = \begin{pmatrix} 2 \\ 3 \end{pmatrix} + \frac{1}{53} \begin{pmatrix} -136 \\ 119 \end{pmatrix} = \frac{1}{53} \begin{pmatrix} -30 \\ -40 \end{pmatrix}$$

$$\Rightarrow F(x_3) = \frac{1}{53} \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} -30 \\ -40 \end{pmatrix} = \frac{1}{53} \begin{pmatrix} 50 \\ -150 \end{pmatrix}$$

$$y_2 = F(x_3) - F(x_2) = \frac{1}{53} \begin{pmatrix} 50 \\ -150 \end{pmatrix} - \begin{pmatrix} 8 \\ -7 \end{pmatrix} = \frac{1}{53} \begin{pmatrix} -374 \\ 221 \end{pmatrix}$$

$$B_3 = B_2 + (y_2 - B_2 s_2) \frac{s_2^T}{s_2^T s_2} = \dots = \frac{1}{1921} \begin{pmatrix} 2425 & -3266 \\ 409 & 4035 \end{pmatrix}$$

Algorithmme BBM :

$$s_1 = -B_1^{-1}F(x_1) = -\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -1 \\ 4 \end{pmatrix} = \begin{pmatrix} 1 \\ -4 \end{pmatrix}$$

$$x_2 = x_1 + s_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} 1 \\ -4 \end{pmatrix} = \begin{pmatrix} 2 \\ -3 \end{pmatrix}$$

$$\Rightarrow F(x_2) = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 2 \\ 3 \end{pmatrix} = \begin{pmatrix} 8 \\ -7 \end{pmatrix}$$

$$y_1 = F(x_2) - F(x_1) = \begin{pmatrix} 8 \\ -7 \end{pmatrix} - \begin{pmatrix} -1 \\ 4 \end{pmatrix} = \begin{pmatrix} 9 \\ -11 \end{pmatrix}$$

$$B_2^{-1} = B_1^{-1} + (s_1 - B_1^{-1}y_1) \frac{y_1^T}{y_1^T y_1} = \dots = \frac{1}{202} \begin{pmatrix} 130 & 88 \\ 63 & 125 \end{pmatrix}$$

$$s_2 = -B_2^{-1}F(x_2) = -\frac{1}{202} \begin{pmatrix} 130 & 88 \\ 63 & 125 \end{pmatrix} \begin{pmatrix} 8 \\ -7 \end{pmatrix} = \frac{1}{202} \begin{pmatrix} -424 \\ 371 \end{pmatrix}$$

$$x_3 = x_2 + s_2 = \begin{pmatrix} 2 \\ -3 \end{pmatrix} + \frac{1}{202} \begin{pmatrix} -424 \\ 371 \end{pmatrix} = \frac{1}{202} \begin{pmatrix} -20 \\ -235 \end{pmatrix}$$

$$\Rightarrow F(x_3) = \frac{1}{202} \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} -20 \\ -235 \end{pmatrix} = \frac{1}{202} \begin{pmatrix} 450 \\ -725 \end{pmatrix}$$

$$y_2 = F(x_3) - F(x_2) = \frac{1}{202} \begin{pmatrix} 450 \\ -725 \end{pmatrix} - \begin{pmatrix} 8 \\ -7 \end{pmatrix} = \frac{1}{202} \begin{pmatrix} -1166 \\ 689 \end{pmatrix}$$

$$B_3^{-1} = B_2^{-1} + (s_2 - B_2^{-1}y_2) \frac{y_2^T}{y_2^T y_2} = \dots = \begin{pmatrix} 0.6269 & 0.4455 \\ 0.1159 & 0.7346 \end{pmatrix}$$

Annexe 3 : second calcul matriciel :

$$\begin{aligned}G(x) &= F^{-1}(x) = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix}^{-1} x = \frac{1}{5} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} x \\x_1 &= \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 4 \end{pmatrix} \\B_1 &= \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix}\end{aligned}$$

Algorithme BGM :

$$B_1 s_1 = -G(x_1) : \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} s_1^1 \\ s_1^2 \end{pmatrix} = -\frac{1}{5} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} -1 \\ 4 \end{pmatrix}$$

$$\Rightarrow s_1 = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

$$x_2 = x_1 + s_1 = \begin{pmatrix} -1 \\ 4 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 4 \end{pmatrix}$$

$$\Rightarrow G(x_2) = \frac{1}{5} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} -2 \\ 4 \end{pmatrix} = \frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix}$$

$$y_1 = G(x_2) - G(x_1) = \frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix} - \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{5} \begin{pmatrix} -3 \\ 1 \end{pmatrix}$$

$$B_2 = B_1 + (y_1 - B_1 s_1) \frac{s_1^T}{s_1^T s_1} = \dots = \frac{1}{5} \begin{pmatrix} 3 & -10 \\ -1 & 15 \end{pmatrix}$$

$$B_2 s_2 = -G(x_2) : \frac{1}{5} \begin{pmatrix} 3 & -10 \\ -1 & 15 \end{pmatrix} \begin{pmatrix} s_2^1 \\ s_2^2 \end{pmatrix} = -\frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix}$$

$$\Rightarrow s_2 = \frac{1}{7} \begin{pmatrix} -18 \\ -4 \end{pmatrix}$$

$$x_3 = x_2 + s_2 = \begin{pmatrix} -2 \\ 4 \end{pmatrix} + \frac{1}{7} \begin{pmatrix} -18 \\ -4 \end{pmatrix} = \frac{1}{7} \begin{pmatrix} -32 \\ 24 \end{pmatrix}$$

$$\Rightarrow G(x_3) = \frac{1}{5} \frac{1}{7} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} -32 \\ 24 \end{pmatrix} = \frac{1}{35} \begin{pmatrix} -48 \\ 56 \end{pmatrix}$$

$$y_2 = G(x_3) - G(x_2) = \frac{1}{35} \begin{pmatrix} -48 \\ 56 \end{pmatrix} - \frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix} = \frac{1}{35} \begin{pmatrix} -62 \\ 14 \end{pmatrix}$$

$$B_3 = B_2 + (y_2 - B_2 s_2) \frac{s_2^T}{s_2^T s_2} = \dots = \frac{1}{425} \begin{pmatrix} 471 & -802 \\ -337 & 1219 \end{pmatrix}$$

Algorithmme BBM :

$$s_1 = -B_1^{-1}G(x_1) = -\frac{1}{5} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$$

$$x_2 = x_1 + s_1 = \begin{pmatrix} -1 \\ 4 \end{pmatrix} + \begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} -2 \\ 4 \end{pmatrix}$$

$$\Rightarrow G(x_2) = \frac{1}{5} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} -2 \\ 4 \end{pmatrix} = \frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix}$$

$$y_1 = G(x_2) - G(x_1) = \frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix} - \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{5} \begin{pmatrix} -3 \\ 1 \end{pmatrix}$$

$$B_2^{-1} = B_1^{-1} + (s_1 - B_1^{-1}y_1) \frac{y_1^T}{y_1^T y_1} = \dots = \frac{1}{25} \begin{pmatrix} 42 & 1 \\ 1 & 3 \end{pmatrix}$$

$$s_2 = -B_2^{-1}G(x_2) = -\frac{1}{25} \begin{pmatrix} 42 & 1 \\ 1 & 3 \end{pmatrix} \frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix} = \frac{1}{25} \begin{pmatrix} -18 \\ -4 \end{pmatrix}$$

$$x_3 = x_2 + s_2 = \begin{pmatrix} -2 \\ 4 \end{pmatrix} + \frac{1}{25} \begin{pmatrix} -18 \\ -4 \end{pmatrix} = \frac{1}{25} \begin{pmatrix} -68 \\ 96 \end{pmatrix}$$

$$\Rightarrow G(x_3) = \frac{1}{5} \frac{1}{25} \begin{pmatrix} 3 & 2 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} -68 \\ 96 \end{pmatrix} = \frac{1}{125} \begin{pmatrix} -12 \\ 164 \end{pmatrix}$$

$$y_2 = G(x_3) - G(x_2) = \frac{1}{125} \begin{pmatrix} -12 \\ 164 \end{pmatrix} - \frac{1}{5} \begin{pmatrix} 2 \\ 6 \end{pmatrix} = \frac{1}{125} \begin{pmatrix} -62 \\ 14 \end{pmatrix}$$

$$B_3^{-1} = B_2^{-1} + (s_2 - B_2^{-1}y_2) \frac{y_2^T}{y_2^T y_2} = \dots = \frac{1}{505} \begin{pmatrix} 743 & 44 \\ 169 & 27 \end{pmatrix}$$

Annexe 4 : troisième calcul matriciel :

$$\begin{aligned}F(x) &= \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} x \\x_1 &= \begin{pmatrix} 1 \\ 1 \end{pmatrix} \\B_1 &= \left(\begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix}^T \right)^{-1} = \frac{1}{5} \begin{pmatrix} 3 & -1 \\ 2 & 1 \end{pmatrix}\end{aligned}$$

Algorithme BGM :

$$B_1 s_1 = -F(x_1) : \frac{1}{5} \begin{pmatrix} 3 & -1 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} s_1^1 \\ s_1^2 \end{pmatrix} = - \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\Rightarrow s_1 = \begin{pmatrix} -3 \\ -14 \end{pmatrix}$$

$$x_2 = x_1 + s_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} -3 \\ -14 \end{pmatrix} = \begin{pmatrix} -2 \\ -13 \end{pmatrix}$$

$$\Rightarrow F(x_2) = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} -2 \\ -13 \end{pmatrix} = \begin{pmatrix} 24 \\ -41 \end{pmatrix}$$

$$y_1 = F(x_2) - F(x_1) = \begin{pmatrix} 24 \\ -41 \end{pmatrix} - \begin{pmatrix} -1 \\ 4 \end{pmatrix} = \begin{pmatrix} 25 \\ -45 \end{pmatrix}$$

$$B_2 = B_1 + (y_1 - B_1 s_1) \frac{s_1^T}{s_1^T s_1} = \dots = \frac{1}{205} \begin{pmatrix} 51 & -377 \\ 205 & 615 \end{pmatrix}$$

$$B_2 s_2 = -F(x_2) : \frac{1}{205} \begin{pmatrix} 51 & -377 \\ 205 & 615 \end{pmatrix} \begin{pmatrix} s_2^1 \\ s_2^2 \end{pmatrix} = - \begin{pmatrix} 24 \\ -41 \end{pmatrix}$$

$$\Rightarrow s_2 = \frac{1}{530} \begin{pmatrix} 697 \\ 7011 \end{pmatrix}$$

$$x_3 = x_2 + s_2 = \begin{pmatrix} -2 \\ -13 \end{pmatrix} + \frac{1}{530} \begin{pmatrix} 697 \\ 7011 \end{pmatrix} = \frac{1}{530} \begin{pmatrix} -363 \\ 121 \end{pmatrix}$$

$$\Rightarrow F(x_3) = \frac{1}{530} \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} -363 \\ 121 \end{pmatrix} = \frac{1}{106} \begin{pmatrix} -121 \\ 0 \end{pmatrix}$$

$$y_2 = F(x_3) - F(x_2) = \frac{1}{106} \begin{pmatrix} -121 \\ 0 \end{pmatrix} - \begin{pmatrix} 24 \\ -41 \end{pmatrix} = \frac{1}{106} \begin{pmatrix} -2'665 \\ 4'346 \end{pmatrix}$$

$$B_3 = B_2 + (y_2 - B_2 s_2) \frac{s_2^T}{s_2^T s_2} = \dots = \frac{1}{1'210'730} \begin{pmatrix} 290'921 & -2'330'017 \\ 1'210'730 & 3'632'190 \end{pmatrix}$$

Algorithmme BBM :

$$s_1 = -B_1^{-1} F(x_1) = - \begin{pmatrix} 1 & 1 \\ -2 & 3 \end{pmatrix} \begin{pmatrix} -1 \\ 4 \end{pmatrix} = \begin{pmatrix} -3 \\ -14 \end{pmatrix}$$

$$x_2 = x_1 + s_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix} + \begin{pmatrix} -3 \\ -14 \end{pmatrix} = \begin{pmatrix} -2 \\ -13 \end{pmatrix}$$

$$\Rightarrow F(x_2) = \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} -2 \\ -13 \end{pmatrix} = \begin{pmatrix} 24 \\ -41 \end{pmatrix}$$

$$y_1 = F(x_2) - F(x_1) = \begin{pmatrix} 24 \\ -41 \end{pmatrix} - \begin{pmatrix} -1 \\ 4 \end{pmatrix} = \begin{pmatrix} 25 \\ -45 \end{pmatrix}$$

$$B_2^{-1} = B_1^{-1} + (s_1 - B_1^{-1} y_1) \frac{y_1^T}{y_1^T y_1} = \dots = \frac{1}{530} \begin{pmatrix} 615 & 377 \\ -205 & 51 \end{pmatrix}$$

$$s_2 = -B_2^{-1} F(x_2) = -\frac{1}{530} \begin{pmatrix} 615 & 377 \\ -205 & 51 \end{pmatrix} \begin{pmatrix} 24 \\ -41 \end{pmatrix} = \frac{1}{530} \begin{pmatrix} 697 \\ 7011 \end{pmatrix}$$

$$x_3 = x_2 + s_2 = \begin{pmatrix} -2 \\ -13 \end{pmatrix} + \frac{1}{530} \begin{pmatrix} 697 \\ 7011 \end{pmatrix} = \frac{1}{530} \begin{pmatrix} -363 \\ 121 \end{pmatrix}$$

$$\Rightarrow F(x_3) = \frac{1}{530} \begin{pmatrix} 1 & -2 \\ 1 & 3 \end{pmatrix} \begin{pmatrix} -363 \\ 121 \end{pmatrix} = \frac{1}{106} \begin{pmatrix} -121 \\ 0 \end{pmatrix}$$

$$y_2 = F(x_3) - F(x_2) = \frac{1}{106} \begin{pmatrix} -121 \\ 0 \end{pmatrix} - \begin{pmatrix} 24 \\ -41 \end{pmatrix} = \frac{1}{106} \begin{pmatrix} -2'665 \\ 4'346 \end{pmatrix}$$

$$B_3^{-1} = B_2^{-1} + (s_2 - B_2^{-1} y_2) \frac{y_2^T}{y_2^T y_2} = \dots = \frac{1}{3'202'780} \begin{pmatrix} 3'632'190 & 2'330'017 \\ -1'210'730 & 290'921 \end{pmatrix}$$

Bibliographie

- [1] Broyden, C.G. [1965] : *A class of methods for solving nonlinear simultaneous equations*, Mathematics of Computation 19, p 577-593.
- [2] Dennis Jr, J.E. ; Schnabel, R.B. [1996] : *Numerical methods for unconstrained optimization and nonlinear equations*, SIAM, p 168-193.
- [3] Martinez, J.M. [2000] : *Practical quasi-Newton methods for solving nonlinear systems*.
- [4] Martinez, J.M. [1979] : *On the order of convergence of Broyden-Gay-Schnabel's method*, Commentationes Mathematicae Universitatis Carolinae 19, p 107-118.
- [5] Martinez, J.M. [1984] : *A quasi-Newton method with modification of one column per iteration*, Computing 33, p 353-362.
- [6] Martinez, J.M. [1987] : *Quasi-Newton methods with factorization scaling for solving sparse nonlinear systems of equations*, Computing 38, p 133-141.
- [7] Martinez, J.M. ; Ochi, L.S. [1982] : *Sobre dois métodos de Broyden*, Matemática Aplicada e Computacional 1, p 135-141.
- [8] Spedicato, E. ; Huang, Z. [1996] : *Numerical experience with Newton-like methods for nonlinear algebraic systems*, Computing, p 68-89.