République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université Larbi Tébessi - Tébessa
Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie
Département : Mathématiques et Informatique

Mémoire de fin d'étude
Pour l'obtention du diplôme de MASTER
Domaine : Mathématiqueset Informatique
Filière : Mathématiques
Option : Equations Differentielles partielles et appliquées

Thème

La convergence des méthodes itératives pour la résolution des systèmes linéaires

Présenté Par :
Haouam Wala ☺ Lemita Nesrine
Devant le jury :

M. Degaichia Hakima MCA Univérsité Larbi Tébessi Président
Mr.Bougherara Lyazid MAA Univérsité Larbi Tébessi Examinateur
Mr. BOUAZIZ Khelifa MCB Univérsité Larbi Tébessi Encadreur

Date de soutenance : 14/06/2020

Résumé

En analyse numérique, pour résoudre les équations algébriques linéaires, il existe plusieurs méthodes possibles certaines sont directes et d'autres itératives. Dans ce travail, nous étudions les conditions de la convergence des méthodes itératives stationnaires et non stationnaires.

Des expériences numériques préliminaires sont données pour présenter une comparaison entre quelques méthodes itératives.

Les mots clés : Système linéaire, matrice, norme, définie positive, Méthode itérative, Jacobi, Gauss-seidel, Relaxation, Gradient, Gradient conjugué, Algorithme, Convergence.

Abstract

In numerical analysis, to solve linear algebraic equations, there are several possible methods: some are direct and others iterative. In this work, we study the conditions for the convergence of stationary and non-stationary iterative methods.

Preliminary numerical experiments are given to present a comparison between some iterative methods.

Keywords: Linear system, matrix, norm, positive definite, Iterative method, Jacobi, Gauss-seidel, Relaxation, Gradient, Conjugate gradient, Algorithm, Convergence.

ملخص

في التحليل العددي ، لحل المعادلات الجبرية الخطية ، هناك العديد من الطرق الممكنة بعضها مباشر والآخر تكراري. في هذا العمل ندرس شروط التقارب بين الطرق التكرارية الثابتة وغير الثابتة.

يتم إعطاء تجارب عددية أولية لتقديم مقارنة بين بعض الطرق التكرارية.

الكلمات المفتاحية :النظام الخطي ، المصفوفة ، القاعدة ، التحديد الإيجابي ، الطريقة التكرارية ، جاكوبي، غوص- صيدل، الاسترخاء ، التدرج المقارن ، الخوارزمية ، التقارب

Remerciments

Nous tenons tout d'abord à remercier **Dieu** le tout puissant, qui nous a donné la force et la patience d'accomplir ce Modeste travail.

Nous tenons à adresser notre remerciements les plus chaleureux et notre profonde gratitude à notre encadreur **Dr.BOUAZIZ Khelifa**, **MCB** à l'Université Elarbi Tebessi- Tebessa pour nos avoir proposé le sujet de ce mémoire. C'est grâce à sa grande disponibilité, ses conseils, ses orientations et ses encouragements que nous avons pu mener à bien ce travail.

Nous sommes conscientes de l'honneur que nous a fait **Dr. Degaichia Hakima, MCA** en étant président du jury et **Dr. Bougherara Lyazid ,MAA** d'avoir accepté d'examiner ce travail.

Nos remerciement s'adresse également à tout nos professeurs pour leurs générosités et la grande patience dont ils ont su faire preuve malgré leurs charges académiques et professionnelles.

Nos profonds remerciements vont également à toutes les personnes qui nous ont aidés et soutenue de prés ou de loin.

Enfin, Merci à tous mes amis et tous mes collègues sans exception, nous n'oublions pas de remercier toutes les personnes qui nos ont facilité la tâche.

الإهداء

إلى من وضعتني على طريق الحياة، جنة الدنيا أمي الغالية.

إلى من أحمل اسمه بكل فخر، ومن افتقده منذ الصغر الغائب عن الدنيا، الحاضر في قلبي أبي طيب الله مثواه.

إلى من كان له بالغ الأثر في كثير من العقبات و الصعاب أخي الحبيب (هاني).

إلى صاحب السيرة العطرة،والفكر المُستنير؛ داعمي وسندي عمي (الميطة التجاني) أطال الله عمره.

إخوتي لعبيدي و عبد الحفيظ، دليلة وعبير.

لا أنسى أغلى صديقاتي إيمان و حبيبة قلبي بلقيس جميع أحبتي و رفيقاتي.

لكل من قال أنَنا لن نصل، ها نحن بفضل الله هُنا.

أُهدي لكم
هذا العمل المتواضع

Dédicace

J'ai l''immense honneur de dédier ce mémoire :

A mes très chers ***parents****qui m'ont soutenu et encourager durant toute ma carrière*

Je dédie ce travail aussi à mes cher frères ***Kheir eddine****,* ***Mohi eddine****,* ***Nadjm Eddine****et mes très chères sœurs* ***Wissal***

- *A toute la famille "* ***Haouam*** *"*
- *A tous mes chers amis*
- *A tous mes chers amis de Université Larbi – tébessa surtout Département :Mathématique et Informatique*

Table des matières

Introduction Générale

En analyse numérique, les méthodes directes de résolution de systèmes linéaires fournissent une solution x au problème $Ax = b$ de la matrice A est élevé, le nombre d'opérations est aussi élevé et de plus, le résultat obtenu n'est pas rigoureusement exat.

Pa railleurs, il existe des cas où les structures du système linéaire ne sont pas tirés à profit par les méthodes directes. C'est par exemple le cas des systèmes où la matrice A est très creuse. C'est la raison pour laquelle, dans ce cas, on préfère utiliser des méthodes itératives.

Une méthode itérative est un procédé algorithmique utilisé pour résoudre un problème, par exemple la recherche d'une solution d'un système d'équations ou d'un problème d'optimisation. En débutant par le choix d'un point initial considéré comme une première ébauche de solution, la méthode procède par itérations au cours desquelles elle détermine une succession de solutions approximatives raffinées qui se rapprochent graduellement de la solution cherchée. Les points générés sont appelés des itérés.

Dans ce mémoire, nous avons de présenter la convergence des plusieurs méthodes itératives pour résoudre les systèmes linéaires des équations algébriques et les comparer entres eux.

Pour atteindre cet objectif, notre travail est articulé autour de trois chapitres.

Le premier chapitre commence par traiter des quelques concepts de base sur les matrices et certains de leurs types. En plus la généralité sur les systèmes linéaires et les normes (normes vectorielles ,normes matricielles).

Dans le deuxième chapitre, nous avons mentionné certains types de méthodes itératives stationnaire et non stationnaire.

Finalement, le dernier chapitre sera consacré aux étude la convergence de les méthodes itératives.

Enfin, nous avons fourni des exemples pour faciliter la comparaison entre les méthodes susmentionnées, prises en charge par le programme Matlab.

Chapitre 1

Concepts fondamentaux

Dans ce chapitre, on définit et on introduit les outils fonctionnels de base nécessaires pour résolutions des systèmes linéaires.

1.1 Quelques notions sur les matrices

Dans toute la suite, $\Bbbk$ désignera le corps des scalaires qui vaudra soit $\mathbb{R}$ pour les réels, soit $\mathbb{C}$ pour les complexes.

Définition 1.1 *Soient $n, m \in \mathbb{N}^*$. Une matrice de type (n, m) sur $\Bbbk$ est un tableau de scalaires (réels ou complexes) à n lignes et m colonnes.*
Ces nombres sont appelés coefficients de la matrice A,

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1j} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2j} & \cdots & a_{2m} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{i1} & a_{i2} & \cdots & a_{ij} & \cdots & a_{im} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nj} & \cdots & a_{nm} \end{pmatrix} \quad \text{ou } A = (a_{ij})_{\substack{1 \le i \le n \\ 1 \le j \le m.}}$$

Nous notons $M_{nm}(\Bbbk)$ l'ensemble des matrices (n, m) sur $\Bbbk$.

Définition 1.2 • *Si $n = m$, nous notons juste $M_n(\Bbbk)$ l'ensemble des matrices carrées (n, n) sur $\Bbbk$. De plus, quand aucune autre notation ne sera précisée, nous noterons a_{ij} le coefficient de la matrice A à l'intersection de la ligne i et de la colonne j.*

$$A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$$

• *Dans une matrice carrée d'ordre n, les coefficients* $a_{11}\ a_{22} \ldots a_{nn}$ *forment la diagonale principale de la matrice.*

• *La matrice unité d'ordre n, notée I_n (matrice identité), est la matrice carrée d'ordre n contenant uniquement des 1 sur sa diagonale principale et 0 ailleurs.*

• *La matrice nulle d'ordre n, notée 0_n , toute matrice dont les coefficients sont tous nuls. Dans ce cas,* $a_{ij} = 0$.

1.1.1 Opérations

Somme

Définition 1.3 *Soient A et B deux matrices ayant la même taille $n \times m$.*
Leur somme $C = A + B$ est la matrice de taille $n \times m$ définie par

$$c_{ij} = a_{ij} + b_{ij}, \text{ pour tous } i = 1, ..., n \text{ et } j = 1, ..., m.$$

En d'autres termes, on somme coefficients par coefficients.

Multiplication par un scalaire

Définition 1.4 *Le produit d'une matrice $A = (a_{ij})$ de $M_{nm}(\Bbbk)$ par un scalaire $\alpha \in \Bbbk$ est la matrice (αa_{ij}) formée en multipliant chaque coefficient de A par α. Elle est notée αA.*

Remarque 1.1 *La matrice $(-1)A$ est l'opposée de A est notée $-A$.*
La différence $A - B$ est définie par $A + (-B)$.

Proposition 1.1 *Soient A, B et C trois matrices appartenant à $M_{nm}(\Bbbk)$. Soient $\alpha,\ \beta \in \Bbbk$ deux scalaires.*
1. $A + B = B + A$, la somme est commutative,
2. $A + (B + C) = (A + B) + C$, la somme est associative,
3. $A + 0 = A$, la matrice nulle est l'élément neutre de l'addition,
4. $(\alpha + \beta) A = \alpha A + \beta A$,
5. $\alpha (A + B) = \alpha A + \alpha B$.

Produit de deux matrices

Définition 1.5 *Soit A une matrice de dimension $n \times p$ et soit B une matrice de dimension $p \times m$. On appelle le produit $A \times B$ la matrice C de dimension $n \times m$ où chaque coefficient c_{ij} est le produit de la $i^{-éme}$ ligne de A par la $j^{-éme}$ colonne de B.*
Autrement dit. Si on pose $A = (a_{ij})_{\substack{1 \leq i \leq n \\ 1 \leq j \leq p}}$ et $B = (b_{ij})_{\substack{1 \leq i \leq p \\ 1 \leq j \leq m}}$,
$A \times B = (c_{ij})$ avec $c_{ij} = \sum_{k=1}^{m} a_{ik} b_{kj}$, pour tous $i = 1, ..., n$ et $j = 1, ..., m$.

Proposition 1.2 *• La multiplication de matrices n'est pas commutative : $A \times B \neq B \times A$.*
• La multiplication de matrices est associative : $(A \times B) \times C = A \times (B \times C) = A \times B \times C$.
• La multiplication des matrices est distributive par rapport à l'addition : $A \times (B + C) = A \times B + A \times C$.
• la puissance $n^{-ième}$ de A est la matrice, notée A^n, égale au produit de n facteurs de A.

Transposée

Définition 1.6 *Si $A \in M_{nm}(\Bbbk)$, la transposée de A notée A^T, est la matrice de $A \in M_{mn}(\Bbbk)$ définie par $(A^T)_{ij} = A_{ji}$ pour tous $i = 1, ..., m$ et $j = 1, ..., n$.*

Proposition 1.3 *Si $A \in M_{nm}(\mathbb{R})$, nous avons les résultats suivants :*
•$(A^T)^T = A$.
•$(A + B)^T = A^T + B^T$,
•$(\lambda A)^T = \lambda A^T, \lambda \in \mathbb{R}$,
•$(AB)^T = B^T A^T$,
•$(A^{-1})^T = (A^T)^{-1}$.

Matrice adjointe

Définition 1.7 *Si $A \in M_{nm}(\Bbbk)$, la matrice adjointe de A, notée A^*, est la matrice de $M_{mn}(\Bbbk)$ définie par $(A^*)_{ij} = \bar{A}_{ji}$ pour tous $i = 1, ..., m$ et $j = 1, ..., n$.*

Remarque 1.2 *Notons que la matrice adjointe d'une matrice A à coefficients complexes est la matrice transposée de la matrice conjuguée de A, autrement dit*

$$A^* = \overline{A^T}. \tag{1.1}$$

Si $A \in M_{nm}(\mathbb{R})$, nous avons les résultats suivants :
•$(A^*)^* = A$.

•$(A+B)^* = A^* + B^*$,
•$(\lambda A)^* = \overline{\lambda} A^*, \lambda \in \mathbb{C}$,
•$(AB)^* = B^* A^*$,

Matrices particulières

Une matrice carrée $A \in M_{nn}(\mathbb{R})$ de coefficients $(a_{ij}) \in \mathbb{k}$ pour tous $i, j = 1, ..., n$. est
1. diagonale : si $a_{ij} = 0$ pour tout $i \neq j$. On note en général $A = diag(a_{11}, a_{22}, ..., a_{nn})$,
2. une matrice diagonale de taille n particulière est la matrice identité I_n dont tous les coefficients 1, autrement dit $I_n = diag(1, 1, ..., 1)$,
3. triangulaire supérieure si $a_{ij} = 0$ pour tout $i > j$,
4. triangulaire inférieure si $a_{ij} = 0$ pour tout $i < j$,
5. symétrique si A est réelle et si $A = A^T$,
6. hermitienne si A est complexe et si $A = A^*$,
7. orthogonale si A est réelle et si $AA^T = A^T A = I_n$,
8. unitaire si A est complexe et si $AA^* = A^* A = I_n$,
9. normale si $AA^* = A^* A$.

Remarque 1.3 *Une matrice à coefficient dans $\mathbb{R}$ (resp. $\mathbb{C}$) est dite symétrique (resp. hermitienne) ssi $A^* = A$ où A^* est la transposée (resp. la transconjuguée) de A.*

Déterminant

Définition 1.8 *Le déterminant d'une matrice carrée (n, n) est noté $\det(A)$,*

$$\det(A) = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix}.$$

Il est défini par la formule suivante :
Pour tout indice i de $\{1, 2, ..., n\}$, on a

$$\det(A) = \sum_{j=1}^{n} (-1)^{i+j} a_{ij} \triangle_{ij}, \tag{1.2}$$

le déterminant $\triangle_{ij}$ d'ordre $n-1$, obtenu en supprimant dans $\det(A)$ la ligne et la colonne de a_{ij}.

Proposition 1.4 *1.* $\det(I_n) = 1$,
2. $\det(A^T) = \det(A)$*,*
3. $\det(A^*) = \overline{\det(A)}$*,*
4. $\det(AB) = \det A \det B$.
5. Pour tout scalaire $\alpha \in \mathbb{k}$*,* $\det(\alpha A) = \alpha^n \det(A)$*.*
6. Le déterminant d'une matrice triangulaire (a fortiori diagonale) est égal au produit de ses termes diagonaux.

Matrice inverse

Définition 1.9 *Une matrice carrée* A *de taille* n *est une matrice inversible lorsqu'il existe une matrice* B *telle que* $A \times B = B \times A = I_n$.
La matrice B*, notée* A^{-1} *est appelée la matrice inverse de* A.

Inverse et opérations

Nous avons les résultats suivants sur les inverses :
1.$(AB)^{-1} = B^{-1}A^{-1}$,
2.$(A^T)^{-1} = (A^{-1})^T$,
3.$(A^*)^{-1} = (A^{-1})^*$.

Théorème 1.1 *Soit* $A \in M_{nn}(\mathbb{k})$*. Les propositions suivantes sont équivalentes*
1. A *est inversible,*
2. $\det(A) \neq 0$*,*
3. $Ax = 0$ *pour seule solution* $x = 0$*,*
4. pour tout $b \in \mathbb{k}^n$*,* $Ax = b$ *possède une unique solution.*

Valeurs propres

soit $A \in M_n(\mathbb{k})$ une matrice carrée de taille n à coefficients. Les valeurs propres de A sont les n racines (distinctes ou confondues) du polynôme
caractéristique de A défini par

$$P_A(\lambda) = det(A - \lambda I_n), \tag{1.3}$$

et sont données par $\lambda_1, \lambda_2, \cdots, \lambda_n$.

Rayon spectral

Définition 1.10 *Le rayon spectral est le plus grand module des valeurs propres de A, noté $\rho(A)$:*

$$\rho(A) = max_{1\leq i\leq n} |\lambda_i| . \tag{1.4}$$

Définition 1.11 *une matrice A est dit à diagonale dominante par lignes si*

$$|a_{ii}| \geq \sum_{j=1,j\neq i}^{n} |a_{ij}| \text{, avec } i = 1, ..., n \tag{1.5}$$

1.1.2 Produit scalaire

Le produit scalaire de deux vecteurs x et y est

$$\langle x, y\rangle = x^T y = \sum_{i=1}^{i=n} x_i y_i. \tag{1.6}$$

Dans le cas de vecteurs complexes, le produit scalaire hermitien est défini par

$$(x, y) \in \mathbb{C}^n \times \mathbb{C}^n \to x^T y = \sum_{i=1}^{i=n} \bar{x}_i y_i, \tag{1.7}$$

ou le surlignage d'une grandeur indique qu'on en considère le conjugué.

Matrices positives

1.**Cas réel** :

Soit A une matrice symétrique réelle dans $M_n(\mathbb{R})$. On peut lui associer une forme quadratique sur $\mathbb{R}^n$.

$$\langle Ax, x\rangle = x^T Ax = \sum_{i,j=1}^{n} a_{ij} x_i x_j.$$

Nous dirons que A est une matrice définie positive si cette forme quadratique vérifie

$$x^T Ax > 0, \text{ pour tout } x \in \mathbb{R}^n, x \neq 0,$$

2.Cas complexe :

Soit A une matrice hermitienne dans $M_n(\mathbb{C})$. On peut lui associer une forme hermitienne sur $\mathbb{C}^n$.

$$\langle Ax, x\rangle = \bar{x}^T Ax = \sum_{i,j=1}^{n} a_{ij} \bar{x}_i x_j.$$

Nous dirons que A est définie positive si cette forme hermitienne vérifie

$$\langle Ax, x\rangle > 0 \text{ pour tout } x \in \mathbb{C}^n, x \neq 0.$$

1.1.3 Normes vectorielles

Il est possible de définir plusieurs normes dans l'espace vectoriel $\mathbb{R}^n$.

Une norme d'un espace vectoriel E est une application $\|.\|$ de E dans $\mathbb{R}^+$qui vérifie les propriétés suivantes :

$$\begin{aligned}
\forall x &\in E, \|x\| = 0 \Leftrightarrow x = 0, \\
\forall \lambda &\in \mathbb{R}, x \in E, \|\lambda x\| = |\lambda| \, \|x\| , \\
\forall x, y &\in E, \|x + y\| \leq \|x\| + \|y\| .
\end{aligned}$$

Dans $\mathbb{R}^+$, les trois normes les plus courantes sont la norme infinie, la norme 1 et la norme euclidienne.

norme infinie :

$$\|x\|_\infty = \max_{1 \leq i \leq n} |x_i| \tag{1.8}$$

norme1 :

$$\|x\|_1 = \sum_{i=1}^{i=n} |x_i| . \tag{1.9}$$

norme 2 ou norme euclidienne :

$$\|x\|_2 = \sqrt{x^T x} = \left(\sum_{i=1}^{i=n} x_i^2 \right)^{\frac{1}{2}} , \tag{1.10}$$

ou $x = (x_1, \cdots, x_n)^T$.

La norme euclidienne est donc définie par le produit scalaire $x^T y$.

1.1.4 Normes matricielles

Définition 1.12 *Soit $\|.\|$ une norme quelconque sur $\mathbb{R}^n$, on appelle norme matricielle subordonnée de la matrice carrée A d'ordre n*

$$\|A\| = \max_{x \in \mathbb{R}^n, \|x\| \neq 0} \frac{\|Ax\|}{\|x\|} = \max_{x \in \mathbb{R}^n, \|x\| = 1} \|Ax\| \tag{1.11}$$

Proposition 1.5 *$a)$ Une norme matricielle subordonnée vérifie les propriétés suivantes :*

$$\begin{aligned}
\|A\| &= 0 \Leftrightarrow A = 0 \;\; \textit{pour tout } A \\
\|\alpha A\| &= |\alpha| \, \|A\| \;\; \textit{pour tout } A, \alpha \in \mathbb{R} \\
\|A + B\| &\leq \|A\| + \|B\| \;\; \textit{pour tout } \; A, B \\
\|AB\| &\leq \|A\| \, \|B\| \;\; \textit{pour tout } A, B
\end{aligned}$$

b) Soit A une matrice carrée réelle, pour les normes vectorielles usuelles on a les normes matricielles subordonnée.

$$\begin{aligned}
\|A\|_\infty &= \max_{1\leq i\leq n}\sum_{j=1}^{n}|a_{ij}| = \textit{maximum sur les lignes} \\
\|A\|_1 &= \max_{1\leq j\leq n}\sum_{i=1}^{n}|a_{ij}| = \textit{ maximum sur les colonnes} \\
\|A\|_2 &= \sqrt{\rho\left(A^TA\right)} = \left\|A^T\right\|_2,
\end{aligned}$$

où $\rho\left(A^TA\right)$ est le rayon spectral de la matrice symétrique A^TA
c) Si O et P sont des matrices orthogonales ($i.e. O^T = O^{-1}$et $P^T = P^{-1}$), alors $\|OAP\|_2 = \|A\|_2$.
Si $AA^T = A^TA$ (i.e. A est normale), alors $\|A\|_2 = \rho(A)$.

1.2 Généralité sur les systèmes linéaires

Définition 1.13 *On appelle système de n équations linéaires à m inconnues (x_j) tout système d'équations de la forme :*

$$\begin{cases}
a_{11}x_1 + a_{12}x_2 + \cdots + a_{1m}x_m = b_1 \\
a_{21}x_1 + a_{22}x_2 + \cdots + a_{2m}x_m = b_2 \\
\cdots \\
a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nm}x_m = b_n,
\end{cases}$$

a_{ij} sont les coefficients de système, b_i sont les composants de seconde nombre (constante) pour $i \in [1,\cdots,n]$ et $j \in [1,\cdots,m]$.
Lorsque tous les b_i sont nuls on dit que le système est homogène.

Un système d'équations linéaires peut aussi s'écrire sous la forme matricielle $Ax = b \Leftrightarrow [A \mid b]$ ou :
A est un matrice de taille $n \times m$.
x est un vecteur de taille m
b est un vecteur de taille n.

$$Ax = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_m \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_n \end{pmatrix},$$

équations les matrice augmentée $[A \mid b]$

$$\left[\begin{array}{cccc|c} a_{11} & a_{12} & \cdots & a_{1m} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2m} & b_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nm} & b_n \end{array}\right]$$

1.2.1 Classification des systèmes linéaires

Il y'a trois cas de système $Ax = b$:

1) Si le nombre d'équations est supérieure au nombre d'inconnues $(n > m)$ (pas de solution)

2) Si le nombre d'inconnues est supérieure au nombre d'équations $(m > n)$ (il existe une infinité de solutions)

3) Si le nombre d'inconnues égal le nombres *d'*équations $(n = m)$,
alors la matrice A est carrée et le système $Ax = b$ admet une solution unique avec un condition $x = A^{-1}b$ tq $\det(A) \neq 0$ si A inversible.

Nous étudierons un seul cas où $n = m$ (A est une matrice carrée).

1.2.2 L'existence et l'unicité de la solution

On a trois conditions assuré l'existence et l'unicité de la solution du système $Ax = b$:

1) A est inversible

2) $rang\ (A) = n$ (le nombre maximale de vecteur ligne (colonne) linéairement indépendant)

3) Le système homogène $Ax = 0$ admet seulement la solution nulle.

Remarque 1.4 *Si la taille de la matrice est assez grand, le calcule de déterminant de A sera très diffcile.*
Dans ce cas, nous avons recourt à ce théorème.

Théorème 1.2 *Soit $A = (a_{ij})$ une matrice carrée d'ordre n. Si A est symétrique défnie positive alors le système $Ax = b$ admet une solution unique.*

Théorème 1.3 *soit A une matrice symétrique. Alors A est définie positive si et seulement si une des propriétés suivantes est satisfaite :*
1. $x^T Ax > 0, \forall x \neq 0$ avec $x \in \mathbb{R}^n$;
2.Les valeurs propres de sous-matrices principales de A sont toutes positives ;
3.Les mineures principaux dominants de A sont tous positifs ;
4.Il existe une matrice inversible H telle que $A = HH^T$;

1.3 Système linéaire triangulaires

1.3.1 Système linéaire triangulaire supérieure

Une matrice U carré et inversible

$\forall j < i$, $U_{ij} = 0$

$$U = \begin{pmatrix} U_{11} & U_{12} & \cdots & U_{1n-1} & U_{1n} \\ 0 & U_{22} & \cdots & U_{2n-1} & U_{2n} \\ 0 & 0 & U_{33} & \ddots & \cdots \\ 0 & 0 & \ddots & \ddots & U_{n-1n} \\ 0 & 0 & 0 & 0 & U_{nn} \end{pmatrix},$$

système triangulaire supérieure :$Ux = b$

$$Ux = b \Leftrightarrow \begin{cases} U_{11}x_1 + U_{12}x_2 + \cdots + U_{1n-1}x_{n-1} + U_{1n}x_n = b_1 \\ U_{22}x_2 + U_{23}x_3 + \cdots + U_{2n-1}x_{n-1} + U_{2n}x_n = b_2 \\ L_{33}x_3 + \cdots + U_{3n-1}x_{n-1} + U_{3n}x_n = b_3 \\ U_{nn}x_n = b_n. \end{cases}.$$

Donc

$$\begin{cases} x_n = \frac{b_n}{U_{nn}}, U_{nn} \neq 0 \\ x_i = \frac{1}{U_{ii}} \left(b_i - \sum_{j=i+1}^{n} U_{ij}x_j \right) \end{cases} \qquad \forall i = n-1 \cdots, 1 \tag{1.12}$$

1.3.2 Système linéaire triangulaire inférieure

Une matrice L carré et inversible

$\forall j > i$, $L_{ij} = 0$

$$L = \begin{pmatrix} L_{11} & 0 & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 & 0 \\ \cdots & \cdots & \ddots & \ddots & 0 \\ L_{n1} & L_{n2} & \cdots & L_{nn-1} & L_{nn} \end{pmatrix},$$

système triangulaire inférieure : $Lx = b$

$$Lx = b \Leftrightarrow \begin{cases} L_{11}x_1 = b_1 \\ L_{21}x_1 + L_{22}x_2 = b_2 \\ L_{31}x_1 + L_{32}x_2 + L_{33}x_3 = b_3 \\ L_{n1}x_1 + L_{n2}x_2 + \cdots + L_{nn-1}x_{n-1} + L_{nn}x_n = b_n. \end{cases}.$$

Donc

$$\begin{cases} x_1 = \frac{b_1}{L_{11}}, L_{11} \neq 0 \\ x_i = \frac{1}{L_{ii}} \left(b_i - \sum_{j=1}^{i-1} L_{ij}x_j \right) \end{cases} \qquad \forall i = 2 \cdots, n \tag{1.13}$$

Chapitre 2

Méthodes itératives pour la résolution de systèmes linéaires

Ce chapitre introduit une classe importante d'algorithmes de résolution des systèmes linéaires par méthodes itératives.

Définition 2.1 *Les méthodes itératives sont particulièrement intéressantes pour les très grandes matrices ou matrices creuse.*
L'idée principale dans les méthodes itératives pour la résolution de système $Ax = b$ est de construire une suite $(x_n)_n$ telle que cette suite doit converge vers la solution exacte x du système linéaire $Ax = b$.

2.1 Le principe de la méthode itérative :

Une technique générale pour résoudre le problème linéaire $Ax = b$, une technique consiste à réécrire la matrice A sous la forme $A = M - (M - A)$, où M est une matrice bien choisie, et d'appliquer le schéma itérative suivant :

$$Mx_n = (M - A)\, x_n + b. \tag{2.1}$$

Quand on se rapproche, on le trouve bien $Ax = (M - (M - A))\, x = (M - N)\, x = b$.
Pour cela, on utilise une suite $x^{(k)}$ qui converge vers un point fixe x.
On cherche à construire l'algorithme pour $x^{(0)}$ donné, la suite $x^{(k+1)} = F\left(x^{(k)}\right)$ avec $k \in \mathbb{N}$.
Lorsque la matrice M est inversible.

$$\begin{aligned} Ax &= b \Leftrightarrow Mx = Nx + b \qquad (2.2)\\ &\Leftrightarrow x = \left(M^{-1}N\right) x + M^{-1}b = F(x), \end{aligned}$$

et F est une fonction affine. La matrice $T = M^{-1}N$ est alors appelée matrice de Jacobi.

Cependant, la matrice A est à diagonale strictement dominante sur les lignes.

Remarquent que cette équation est équation de la forme $x = F(x)$.

Par conséquant, les méthodes itératives sont des méthodes de poin fixe.

La détermination du point fixe repose sur la détermination de l'équation

$$x^{k+1} = \left(M^{-1}N\right) x^k + M^{-1}b,$$

en notons $x^{(k)}$ le vecteur de composantes $x^{(k)} = \left(x_1^{(k)}, x_2^{(k)}, \cdots, x_n^{(k)}\right)^t$.

L'algorithme est initialisée par un vecteur arbitrair $x^{(0)} = \left(x_1^{(0)}, x_2^{(0)}, \cdots, x_n^{(0)}\right)^t$, et s'arrête quand :

$$\left|x_i^{(k+1)} - x_i^{(k)}\right| \leq \varepsilon \text{ pour un } \varepsilon \text{ donné.}$$

En posant $T = M^{-1}N$, et $c = M^{-1}b$.

Alors la méthode itérative s'écrit sous la forme :

$$\begin{cases} x^{(0)} \text{ vecteur de départ} \\ x^{(k+1)} = Bx^{(k)} + c \quad \forall k \geq 0 \text{ sinon.} \end{cases} \tag{2.3}$$

Où T désigne une matrice carrée appelée matrice d'itération et c un vecteur dépendant de b (le second membre du système à résoudre).

Remarque 2.1 *Les méthodes itératives peut définir par (2.3) .*

Si la suite de solutions approchées $x^{(k)}$ converge vers une limite x quand k tend vers l'infini, alors par passage au limite dans la relation de récurrence (2.3), on obtient

$$(M - N)\, x = Ax = b.$$

Par conséquent, si la suite des solutions approchées converge, sa limite est forcément la solution du système linéaire.

D'un point de vue pratique, il faut savoir quand on peut arrêter les itérations, c'est- à-dire à quel moment $x^{(k)}$ est suffisamment proche de la solution inconnue x.

Comme on ne connait pas x, on ne peut pas décider d'arrêter le calcule lorsque $\left\|x - x^{(k)}\right\| \leq \varepsilon$ ou ε est la précision désirée.

Par contre on connait Ax (qui vaut b), est un critère d'arrêt fréquement utilisé est :

$$\left\|b - Ax^{(k)}\right\| \leq \varepsilon.$$

Cependant, si la norme de A^{-1} est grande ce critère peut être trompeur car :

$$\left\|x - x^{(k)}\right\| \leq \left\|A^{-1}\right\| \left\|b - Ax^{(k)}\right\| \leq \varepsilon \left\|A^{-1}\right\|,$$

qui peut ne pas être petit.

On suuppose donnée A avec $a_{ii} \neq 0 \ \forall i$. On décompose A sous la forme :

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & \cdots & \cdots & a_{1n} \\ a_{21} & a_{22} & & & -F & \\ & & & D & & \\ & -E & & & & \\ a_{n1} & \cdots & \cdots & \cdots & a_{nn-1} & a_{nn} \end{bmatrix} = D - E - F$$

Notation 2.1 *La matrice A va s'écrire $A = D - E - F$, où D, E et F sont des matrices de la même taille que A et telles que :*

• *D : Matrice ne contenant que les termes diagonaux de A*

$$D = \begin{pmatrix} a_{11} & 0 & \cdots & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & . \\ \cdots & . & . & . & 0 \\ \cdots & . & . & . & 0 \\ 0 & . & . & 0 & a_{nn} \end{pmatrix}$$

• *$(-E)$: Matrice ne contenant que l'opposé des coefficients de partie triangulaire inférieure de A de diagonale nulle.*

$$-E = \begin{pmatrix} 0 & 0 & \cdots & \cdots & 0 \\ a_{21} & 0 & . & \cdots & \cdots \\ . & . & . & \cdots & . \\ . & . & . & . & . \\ a_{n1} & . & . & a_{nn-1} & 0 \end{pmatrix}$$

• *$(-F)$: Matrice ne contenant que l'opposé des coefficients de partie triangulaire supérieur de A de diagonale nulle .*

$$-F = \begin{pmatrix} 0 & a_{12} & \cdots & \cdots & a_{1n} \\ 0 & 0 & \cdots & \cdots & \cdots \\ . & . & . & \cdots & \cdots \\ . & . & . & . & a_{n-1n} \\ 0 & \cdots & . & . & 0 \end{pmatrix}$$

2.2 Description des méthodes itératives

2.2.1 Méthode de Jacobi

La méthode de Jacobi, est une méthode itérative de résolution d'un système matriciel de la forme $Ax = b$.

Pour cela, on utilise une suite $x^{(k)}$ qui converge vers un point fixe x.

Pour la méthode de Jacobi, on écrit chaque inconnue en fonction des autres inconnues, nous illustrons cette idée par utilisée un système $n \times n$:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \cdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}.$$

On remarque que si les éléments diagonaux de A sont non nuls, le système linéaire $Ax = b$ est équivalent à :

$$\begin{cases} x_1 = \left[b_1 - (a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n)\right]/a_{11} \\ x_2 = \left[b_2 - (a_{21}x_1 + a_{23}x_3 + \cdots + a_{2n}x_n)\right]/a_{22} \\ \cdots \\ x_n = \left[b_n - (a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn-1}x_{n-1})\right]/a_{nn}. \end{cases}.$$

On suppose une valeur initiale pour $x_1, x_2, \cdots x_n$. On l'insère dans la coté droite de l'égalité et calculer la coté seconde des valeurs approximés des inconnues.

Les nouvelles valeurs son substituée on la coté droite de l'égalité pour obtenu la troisième ensemble de la solution approchée,...etc.

L'itération est défini par :

$$x_i = 1/a_{ii}\left(b_i - \sum_{\substack{j=1, \\ j\neq i}}^{n} a_{ij}x_j\right).1 \leq i \leq n$$

Dans la méthode de Jacobi, on choisit $M = D$ et $N = E + F$

$$Ax = b \Leftrightarrow Dx = (E + F)\,x + b,$$

et à la méthode itérative de Jacobi :

$$\begin{cases} x^0 \quad \text{arbitraire} \\ x^{(k+1)} = D^{-1}(E + F)\,x^{(k)} + D^{-1}b \end{cases}.$$

$x_i^{(k+1)} = \cdots x_i^{(k)} + \frac{b_i}{a_{ii}}$ avec pour la ligne i de $D^{-1}(E+F)$: $-\left(\frac{a_{i1}}{a_{ii}}, \cdots, \frac{a_{ii-1}}{a_{ii}}, 0, \frac{a_{ii+1}}{a_{ii}}, \cdots, \frac{a_{in}}{a_{ii}}\right)$, on a alors :

$$x_i^{(k+1)} = -\frac{1}{a_{ii}} \sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} x_j^{(k)} + \frac{b_i}{a_{ii}}$$

Vecteur résidu

Soit $r^{(k)} = De^{(k)}$ le vecteur résidu. On peut écrire $x_i^{(k+1)} = \frac{r_i^{(k)}}{a_{ii}} + x_i^{(k)}$avec $r_i^{(k)}$ que l'on calcule de la manière suivante :

$$r_i^{(k+1)} = -\sum_{\substack{j=1 \\ j\neq i}}^{n} a_{ij} \frac{r_i^{(k)}}{a_{jj}}.$$

Test d'arrêt

Pour le test d'arrêt, on utilise l'erreur relative sur le vecteur résidu, ce qui donne, pour une précision donnée ε :

$$\frac{\left\| r^{(k)} \right\|}{b} < \varepsilon$$

$$\begin{cases} x^{(0)} & & \text{donné} \\ x_i^{(k+1)} = \frac{b_i - \sum_{j\neq i} a_{ij} x_j^{(k)}}{a_{ii}} & \forall i = 1, ..., n & \text{alors que } a_{ii} \neq 0. \end{cases}$$

2.2.2 Méthode de Gauss-Seidel

La méthode de Gauss-Seidel est une méthode itérative de résolution d'un système linéaire (de dimension finie) de la forme $Ax = b$, ce qui signifie qu'elle génère une suite qui converge vers une solution de cette équation, lorsque les conditions de convergence sont satisfaites (par exemple lorsque A est symétrique définie positive).

L'algorithme suppose que la diagonale de A est formée d'éléments non nuls.

On décompose la matrice A de la façon suivante :

$$A = D - E - F, \tag{2.4}$$

avec

- (D) la diagonale
- $(-E)$ la partie en dessous de la diagonale
- $(-F)$ la partie au dessus.

Dans la méthode de Gauss-Seidel, on choisit $M = D - E$ et $N = F$

$$x^{(k+1)} = (D - E)^{-1} F x^{(k)} + (D - E)^{-1} b, \tag{2.5}$$

on a alors :

$$x_i^{(k+1)} = \frac{1}{a_{ii}} / \left(b_{i-} \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right), 1 \leq i \leq n$$

La matrice $T_{GS} = (D - E)^{-1} F$ est dite matrice de Gauss-Seidel associée à la matrice A.

Remarque 2.2 *Si D est inversible, la matrice de Gauss-Seidel s'écrit*

$$T_{GS} = (I - D^{-1}E)^{-1} D^{-1} F.$$

$$\begin{cases} x^{(0)} & \text{donné} \\ x_i^{(k+1)} = \frac{b_i - \sum\limits_{j<i} a_{ij} x_j^{(k+1)} - \sum\limits_{j>i} a_{ij} x_j^{(k)}}{a_{ii}} \quad \forall i = 1, \ldots, n & \text{alors que } \; a_{ii} \neq 0 \end{cases}$$

2.2.3 Méthode de relaxation

On fait la méme hypothèse que pour les autres mèthodes, c'est-à-dire que D soit inversible. On introduit un paramètre réel ω non nul. Ceci implique que $\frac{D}{\omega} - E$ est aussi inversible.

On écrit $Ax = b$ sous la forme générale où $M = \frac{1}{\omega} D - E$, et $N = \frac{1-\omega}{\omega} D + F$,

$$(\frac{D}{\omega} - E)x = (F + \frac{1-\omega}{\omega} D)x + b,$$

ceci implique

$$x = (\frac{D}{\omega} - E)^{-1}(F + \frac{1-\omega}{\omega} D)x + (\frac{D}{\omega} - E)^{-1} b, \tag{2.6}$$

qui est de la forme $x = Tx + c$ si on choisit

$$T = (\frac{D}{\omega} - E)^{-1}(F + \frac{1-\omega}{\omega} D)$$
$$c = (\frac{D}{\omega} - E)^{-1} b$$

$$\begin{cases} x_0 \in \mathbb{k}^n & \text{donné,} \\ \left(\frac{1}{\omega} D - E\right) x_{k+1} = \left(\frac{1-\omega}{\omega} D + F\right) x_k + b, & \text{pour tout entier } k \geq 0. \end{cases}$$

2.3 Méthodes itratives non stationnaire

Définition 2.2 *Soit $\Omega \subset \mathbb{R}^n$ un **ouvert** et $f : \Omega \longrightarrow \mathbb{R}^n$.*
1. On dit que f est de classe C^m sur Ω ($f \in C^m(\Omega)$) si toutes les dérivées partielles jusqu'à l'ordre m existent et sont continues.
2. Pour tout $x \in \Omega$ et tout $i \in \{1, 2, \ldots, n\}$ on note

$$\frac{\partial f}{\partial x_i}(x) = \lim_{t \longmapsto 0} \frac{1}{t}[f(x + tx_i) - f(x)],$$

*(c'est la **dérivée partielle** de $f$en x de direction x_i.)*
3. Pour tout $x \in \Omega$ on note

$$\nabla f(x) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n}\right)^T \in \mathbb{R}^n, \quad \forall x \in \Omega.$$

*(le **gradient** de f en x). $\nabla f = g^T$.*
4. Pour tous $x \in \Omega$ et $h \in \mathbb{R}^n$ on note

$$\frac{\partial f}{\partial h}(x) = \lim_{t \mapsto 0} \frac{1}{t}[f(x + th) - f(x)] = g'(0).$$

*(c'est la **dérivée directionnelle** de f en x de direction h) où on a noté $g(t) = f(x + th)$.*

Remarque 2.3 *Nous rappellons aussi la formule :*

$$\frac{\partial f}{\partial h}(x) = \langle \nabla f(x), h \rangle, \quad \forall x \in \Omega \;\; \forall h \in \mathbb{R}^n.$$

Nous étudions un autre type des méthodes itératives pour résoudre numériquement $Ax = b$, le vecteur b étant donné.

Commençons par définir, pour tout N-vecteur y, la quantité :

$$P(y) = \frac{1}{2}\langle y, Ay \rangle - \langle b, y \rangle = \frac{1}{2} y^t Ay - b^t y. \tag{2.7}$$

Il est clair que $P(y)$ est un nombre réel, et P est une application de $\mathbb{R}^n$ dans $\mathbb{R}$.

Théorème 2.1 *Si A est une $n \times n$ matrice symétrique définie positive et si x est solution de $Ax = b$ alors, pour tout n-vecteur y différent de x on a :*

$$P(x) < P(y).$$

Preuve. [10]. Soit x tel que $Ax = b$ et soit y un vecteur différent de x. Par hypothèse, le vecteur $z = x - y$ est donc différent de zéro. Par définition de P nous avons :

$$\begin{aligned} P(y) &= P(x-z) = \frac{1}{2}(x-z)^t A(x-z) - b^t(x-z). \\ &= \frac{1}{2}x^t Ax - \frac{1}{2}x^t Az - \frac{1}{2}z^t Ax + \frac{1}{2}z^t Az - b^t x + b^t z. \end{aligned}$$

En tenant compte de la symétrie de A, nous avons $x^t Az = z^t Ax$ ainsi que $b^t z = z^t b$, si bien que

$$\begin{aligned} P(y) &= P(x) - z^t Ax + z^t b + \frac{1}{2}z^t Az \\ &= P(x) - z^t(Ax - b) + \frac{1}{2}z^t Az, \end{aligned}$$

et puisque $Ax = b$, nous obtenons

$$P(y) = P(x) + \frac{1}{2}z^t Az.$$

La matrice A étant symétrique définie positive et le vecteur z étant non nul, nous avons $z^t Az > 0$ et ainsi $P(y) > P(x)$. ■

La solution du système $Ax = b$ réalise le minimum de P. Nous allons construire une méthode itérative qui chercher, à chaque itération, à diminuer P.

Supposons donc que l'on a obtenue $x^{(k)}$ et calculons $x^{(k+1)}$ de sorte à ce que :

$$P\left(x^{(k+1)}\right) < P\left(x^{(k)}\right).$$

Ce qui revient à déterminer des directions de déplacement qui permettent de se rapprocher le plus possible de la solution.

Une idée légitime consiste à choisir un vecteur $d^{(k)}$ non nul et à poser

$$x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)}. \tag{2.8}$$

Où α_k est un nombre réel qui minimise la quantité $F(\alpha)$ définie par :

$$F(\alpha) = P\left(x^{(k)} + \alpha d^{(k)}\right). \tag{2.9}$$

Cette manière de calcule $x^{(k+1)}$ à partir de $x^{(k)}$ est appelée méthode de descente, le vecteur $d^{(k)}$ est appelé **direction de descente**.

Explicitons maintenant le calcule de α_{k+1} tel que $F(\alpha_{k+1}) \leq F(\alpha_k)$, $\forall \alpha \in \mathbb{R}$.

Nous annulons la dérivée $F'(\alpha)$. En utilisant (2.7).

Nous avons

$$P(y) = \frac{1}{2}\sum_{i,j=1}^{n} a_{ij} y_i y_j - \sum_{i=1}^{n} b_i y_i, \tag{2.10}$$

et par suite, en tenant compte de la symétrique de A :

$$\frac{\partial}{\partial y_i} P(y) = \sum_{i,j=1}^{n} a_{ij} y_i - \sum_{i=1}^{n} b_i$$

Il sufft d'utiliser la définition (2.9) pour montre que

$$F'(\alpha_{k+1}) = \left(d^{k+1}\right)^t \left(A\left(x^{k+1} + \alpha_{k+1} d^{k+1}\right) - b\right). \tag{2.11}$$

Clairement, si nous voulons que α_{k+1}soit tel que $F'(\alpha_{k+1}) = 0$, nous déduisons de (2.11) que α_{k+1} est donner par :

$$\alpha_{k+1} = \frac{\left(d^{k+1}\right)^T \left(b - Ax^{k+1}\right)}{\left(d^{k+1}\right)^T A d^{k+1}}. \tag{2.12}$$

En déffnissant encore

$$g = Ax - b,$$

qui est le résidu à l'étape n, la méthode de descente peut se résumer ainsi lorsqu'on veut calculer $x^{(k+1)}$ à partire de $x^{(k)}$:

- On choisit une direction de descente d^{k+1} ;
- On calcule

$$\alpha_{k+1} = -\frac{\left(d^{k+1}\right)^T g^{(k+1)}}{\left(d^{k+1}\right)^T A d^{k+1}}$$

- On calcule $x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)}$.

2.3.1 Méthode du Gradient

Pour la méthode du gradient on choisit d égale à la direction de la plus grande pente de P au point $x^{(k)}$. Intuitivement, cela permet d'espérer une réduction signi cative d'énergie pour $P(x^{(k+1)})$.

La direction d est donc donnée par :

$$d^{(k)} = -\nabla P\left(x^{(k)}\right) = Ax^{(k)} - b.$$

Ou bien consiste à faire les itérations suivantes

$$\begin{cases} d^k = -\nabla P(x^k), \\ x^{k+1} = x^k + \alpha_k d^k. \end{cases} \tag{2.13}$$

Où α_k est choisi de manière à ce que

$$P(x^k + \alpha_k d^k) \leq P(x^k + \alpha d^k), \ \forall \alpha > 0. \tag{2.14}$$

On obtient alors la méthode du gradient à pas optimal, cette méthode possède une propriété interessante :

Proposition 2.1 *Soit $P : \mathbb{R}^n \to \mathbb{R}$ une fonction différentiable. Les directions de descente d^k générées par la méthode (2.13) et (2.14) vérifient*

$$(d^{k+1})^T d^k = 0. \tag{2.15}$$

Preuve. Si on introduit la fonction

$$F(\alpha) = P(x^k + \alpha d^k),$$

on a

$$F'(\alpha) = \nabla^T P(x^k + \alpha d^k) d^k,$$

et puisque F est dérivable on a nécessairement

$$F'(\alpha) = 0,$$

donc

$$\nabla^T P(x^k + \alpha d^k) d^k = \nabla^T P(x^{k+1}) d^k = -(d^{k+1})^T d^k = 0.$$

■

Pour A symétrique, définie positive.

Notons que le facteur devant d importe peu, il est corrigé par α_k .

Tq : α_k est le paramètre d'optimalité de la nouvelle solution dans la direction $d^{(k)}$.

Algorithme (du gradient) :

$$d^{(0)} = b - Ax^{(0)}.$$

répéter jusqu'à l'arrêt

$$d^{(k)} = -g^{(k)}$$

$$\begin{cases} \text{calculer} \quad \alpha_k = -\frac{d^{(k)T} g^{(k)}}{d^{(k)T} A d^{(k)}} \\ x^{(k+1)} = x^{(k)} + \alpha_k d^{(k)} \\ g^{(k+1)} = g^{(k)} - \alpha_k A d^{(k)} \end{cases}$$

Notation 2.2 *Comme avant, la multiplication $Ad^{(k)}$ ne se fait qu'une fois par itération.*

Remarque 2.4 *le dénominateur de l'expression pour α_k est nul si $\left\|d^{(k)}\right\| A = 0$ et donc $g^{(k)} = 0$, mais alors $x^{(k)}$ est la solution exacte.*

Exemple 2.1 *Considérons la fonction $f : \mathbb{R}^2 \to \mathbb{R}$ définie par*

$$\min f(x_1, x_2) = \min \left\{ \frac{1}{5}x_1^2 + x_2^2,\ (x_1, x_2) \in \mathbb{R}^2 \right\}. \tag{2.16}$$

En démarrant du point initial $x^{(0)} = \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$, et en appliquant la méthode du gradient , trouver la solution optimale $\hat{x} = \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \end{pmatrix}$ du problème (2.16)

$$\begin{aligned} f(x_1, x_2) &= \frac{1}{5}x_1^2 + x_2^2,\ (x_1, x_2) \in \mathbb{R}^2 \\ &= \frac{1}{2}(x_1, x_2) \begin{pmatrix} \frac{2}{5} & 0 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - (x_1, x_2) \begin{pmatrix} 0 \\ 0 \end{pmatrix} \\ &= \frac{1}{2}x^T A x - b^T x. \end{aligned}$$

Calcul de α_0

On a

$$\alpha_k = \frac{\left\|d^k\right\|^2}{\left(d^k\right)^T A d^k}$$

$$d^k = -\nabla^T f(x) = -(\frac{2}{5}x_1, 2x_2) \Rightarrow \left\|d^k\right\|^2 = \frac{4}{25}x_1^2 + 4x_2^2,$$

et

$$\begin{aligned} \left(d^k\right)^T A d^k &= (\frac{2}{5}x_1, 2x_2) \begin{pmatrix} \frac{2}{5} & 0 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} \frac{2}{5}x_1 \\ 2x_2 \end{pmatrix} \\ &= \frac{8}{125}x_1^2 + 8x_2^2. \end{aligned}$$

Donc

$$\alpha_0 = \frac{\frac{4}{25} + 4}{\frac{8}{125} + 8} \simeq 0.5158$$

Calcul de $(x_1^{(1)}, x_2^{(1)})$

$$\begin{aligned} (x_1^{(1)}, x_2^{(1)}) &= (x_1^{(0)}, x_2^{(0)}) + \alpha_0 d^0 \\ &= (1, 1) - 0.5158(\frac{2}{5}, 2) \\ &= (0.79, -0.036). \end{aligned}$$

Calcul de α_1

$$\begin{aligned}
\alpha_1 &= \frac{\|d^1\|^2}{(d^1)^T A d^1} \\
&= \frac{\frac{4}{25}x_1^2 + 4x_2^2}{\frac{8}{125}x_1^2 + 8x_2^2} \\
&= \frac{\frac{4}{25}(0.79)^2 + 4(-0.036)^2}{\frac{8}{125}(0.79)^2 + 8(-0.036)^2} \simeq 2.090147.
\end{aligned}$$

Calcul de $(x_1^{(2)}, x_2^{(2)})$

$$\begin{aligned}
(x_1^{(2)}, x_2^{(2)}) &= (x_1^{(1)}, x_2^{(1)}) + \alpha_1 d^1 \\
&= (0.79, -0.036) - 2.090147 \left[\frac{2}{5}(0.79), 2(-0.036)\right] \\
&= (0.12997, 0.11449).
\end{aligned}$$

2.3.2 La méthode du gradient conjugué

L'algorithme du gradient conjugué construit sa direction de descente en x en ajoutant à l'opposé du gradient, la direction d^{k-1} multipliée par un scalaire $\beta_k \in \mathbb{R}$:

$$d^k = -\nabla P(x) + \beta_k d^{k-1}.$$

2.3.3 Le principe général d'une méthode à directions conjuguées

Définition 2.3 *Soit A une matrice symétrique $n \times n$, définie positive. On dit que deux vecteurs x et y de $\mathbb{R}^n$ sont A-conjugués (ou conjugués par rapport à A) s'ils vérifient $x^T A y = 0$.*

Description de la méthode

Soit $\{d_0, d_1, \ldots, d_n\}$ une famille de vecteurs $A-$conjugués. On appelle alors méthode de directions conjuguées toute méthode itérative appliquée à une fonction quadratique strictement convexe de n variables : $P(x) = \frac{1}{2}x^T Ax - b^T x$, avec $x \in \mathbb{R}^n$ et $A \in M_{n \times n}$ est symétrique et définie positive, $b \in \mathbb{R}^n$ conduisent à l'optimum en n étapes au plus. Et cette méthode de la forme :

$$\begin{aligned}
& x_0 \text{ donné}, \\
x^{k+1} &= x^k + \alpha_k d^k,
\end{aligned} \tag{2.17}$$

où α_k est optimal et $d^0, d^1, \ldots, d^n$ possédant la propriété d'être mutuellement conjuguées par rapport à la fonction quadratique. Si l'on note $g^k = \nabla P(x^k)$, la méthode se construit comme suit :

Calcul de α_k

Comme α_k minimise P dans la direction d^k, on a, $\forall k$:

$$\begin{aligned} F'(\alpha_k) &= d^{(k)^T} \nabla P\left(x^{k+1}\right) \\ &= d^{(k)^T}\left(Ax^{k+1} - b\right). \end{aligned}$$

Soit

$$d^{(k)^T} A(x^k + \alpha_k d^k) - d^{(k)^T} b = 0,$$

d'où l'on tire :

$$\alpha_k = \frac{-d^{(k)^T}(Ax^k - b)}{d^{(k)^T} A d^k}. \tag{2.18}$$

Construire les directions A-conjuguées

Les directions A-conjuguées $d^0, d^1, \ldots, d^n$ peuvent être générées à partir d'un ensemble de vecteurs linéairement indépendants $\xi_0, \ldots, \xi_k$ en utilisant la procédure dite de Gram-Schmidt, de telle sorte que pour tout i entre 0 et k, le sous-espace généré par $d^0, d^1, \ldots, d^i$ soit égale au sous-espace généré par $\xi_0, \ldots, \xi_i$. Alors d^{i+1}est construite comme suit :

$$d^{i+1} = \xi_{i+1} + \sum_{m=0}^{i} \varphi_{(i+1)m} d^m.$$

La méthode du gradient conjugué est obtenue en appliquant la procédure de Gram-Schmidt aux gradients $\nabla P(x^0), \ldots, \nabla P(x^{k-1})$, c'est-à-dire en posant $\xi_0 = -\nabla P(x^0), \ldots, \xi_{k-1} = -\nabla P(x^{k-1})$. En outre, nous avons que

$$\begin{aligned} \nabla P(x) &= Ax - b, \\ \text{et} \quad \nabla^2 P(x) &= A. \end{aligned}$$

Notons que la méthode se termine si $\nabla P(x^k) = 0$. La particularité intéressante de la méthode du gradient conjugué est que le membre de droite de l'équation donnant la valeur de d^{k+1} dans la procédure de Gram-Schmidt peut être grandement simplifié. Notons que la méthode du gradient conjugué est inspirée de celle du gradient (plus profonde pente).

Algorithme de La méthode du gradient conjugué pour les fonctions quadratiques

On suppose ici que la fonction à minimiser est quadratique sous la forme :

$$P(x) = \frac{1}{2} x^T A x - b^T x.$$

Si l'on note $g^k = \nabla P(x^k)$, l'algorithme prend la forme suivante. Cet algorithme consiste à générer une suite d'itérés $\left\{x^k\right\}$ sous la forme :

$$x^{k+1} = x^k + \alpha_k d^k. \tag{2.19}$$

L'idée de la méthode est :

1-construire itérativement des directions $d^0, \ldots, d^k$mutuellement conjuguées

A chaque étape k la direction d^k est obtenue comme combinaison linéaire du gradient en x^k et de la direction précédente d^{k-1} c'est-à-dire

$$d^{k+1} = -\nabla P\left(x^{k+1}\right) + \beta_{k+1} d^k, \tag{2.20}$$

les coeffcients β_{k+1} étant choisis de telle manière que d_k soit conjuguée avec toutes les directions précédentes. Autrement dit :

$$d^{(k+1)^T} A d^{(k)} = 0,$$

on en déduit :

$$\begin{aligned} d^{(k+1)^T} A d^{(k)} &= 0 \Rightarrow \left(-\nabla P\left(x^{k+1}\right) + \beta_{k+1} d^k\right)^T A d^{(k)} = 0 \\ &\Rightarrow -\nabla^T P\left(x^{k+1}\right) A d^{(k)} + \beta_{k+1} d^{(k)^T} A d^{(k)} = 0 \\ &\Rightarrow \beta_{k+1} = \frac{\nabla^T P\left(x^{k+1}\right) A d^{(k)}}{d^{(k)^T} A d^{(k)}} = \frac{g^{(k+1)^T} A d^k}{d^{(k)^T} A d^{(k)}}. \end{aligned} \tag{2.21}$$

Algorithme (gradient conjugué)

Etape 0 : (initialisation)

Soit x^0 le point de depart, $g^0 = \nabla P(x^0) = Ax^0 - b$, poser $d^0 = -g^0$.

Poser $k = 0$ et aller à l'étape 1.

Etape 1 :

Si $g^k = 0$: STOP ($x^* = x^k$)."Test d'arrêt".

Si non aller à l'étape 2.

Etape 2 :

Définir $x^{k+1} = x^k + \alpha_k d^k$ avec :

$$\begin{aligned} \alpha_k &= \frac{-d^{(k)^T} g^k}{d^{(k)^T} A d^{(k)}}, \\ d^{k+1} &= -g^{k+1} + \beta_{k+1} d^k, \\ \beta_{k+1} &= \frac{g^{(k+1)^T} A d^k}{d^{(k)^T} A d^{(k)}}. \end{aligned}$$

Poser $k = k + 1$ et aller à l'étape 1.

La validité de l'algorithme du gradient conjugué

On va maintenant montrer que l'algorithme ci-dessus définit bien une méthode de directions conjuguées.

Théorème 2.2 *A une itération k quelconque de l'algorithme où l'optimum de $P(x)$ n'est pas encore atteint (c'est-à-dire $g^i \neq 0, i = 0, 1, \cdots, k$) on a*

$a)$

$$\alpha_k = \frac{g^{(k)^T} g^k}{d^{(k)^T} A d^{(k)}} \neq 0 \tag{2.22}$$

$b)$

$$\begin{aligned} \beta_{k+1} &= \frac{g^{(k+1)^T}\left[g^{(k+1)} - g^k\right]}{g^{(k)T} g^k} &(2.23)\\ &= \frac{g^{(k+1)^T} g^{(k+1)}}{g^{(k)T} g^k} &(2.24)\end{aligned}$$

$c)$ *Les directions $d^0, d^1, \cdots, d^k$ engendrées par l'algorithme sont mutuellement conjuguées.*

Preuve. [8]

On raisonne par récurrence sur k en supposant que $d^0, d^1, \cdots, d^k$ sont mutuellement conjuguées

$a)$ Montrons d'abord l'équivalence de (2.20) et de (2.22).

On a : $d^k = -g^k + \beta_k d^{k-1}$.

Danc :

$$\begin{aligned} \alpha_k &= \frac{-d^{(k)^T} g^k}{d^{(k)^T} A d^{(k)}} \\ &= \frac{-\left[-g^k + \beta_k d^{k-1}\right]^T g^k}{d^{(k)^T} A d^{(k)}} \\ &= \frac{g^{(k)^T} g^k}{d^{(k)^T} A d^{(k)}} - \beta_k \frac{d^{(k-1)^T} g^k}{d^{(k)^T} A d^{(k)}}. \end{aligned}$$

Comme $(d^0, d^1, \cdots, d^{k-1})$ sont mutuellement conjuguées, x^k est l'optimum de $P(x)$ sur la variété v^k passant par x^0 et engendrée par $(d^0, d^1, \cdots, d^{k-1})$.

Donc $d^{(k-1)^T} g^k = 0$ d'où l'on déduit (2.22).

$b)$ Pour démontrer (2.23) remarquons que :

$$\begin{aligned} g^{k+1} - g^k &= A(x^{k+1} - x^k) = \alpha_k A d^{(k)} \\ &\Rightarrow A d^{(k)} = \frac{1}{\alpha_k}\left[g^{k+1} - g^k\right]. \end{aligned}$$

On a alors :

$$g^{(k+1)^T} A d^{(k)} = \frac{1}{\alpha_k} g^{(k+1)^T}\left[g^{k+1} - g^k\right],$$

et en utilisant (2.22)

$$\alpha_k = \frac{g^{(k)^T} g^k}{d^{(k)^T} A d^{(k)}},$$

il vient

$$\begin{array}{rcl} g^{(k+1)^T}Ad^{(k)} & = & \dfrac{d^{(k)^T}Ad^{(k)}}{g^{(k)^T}g^k} g^{(k+1)^T}\left[g^{k+1}-g^k\right] \\ & \Rightarrow & \dfrac{g^{(k+1)^T}Ad^{(k)}}{d^{(k)^T}Ad^{(k)}} = \dfrac{g^{(k+1)^T}\left[g^{k+1}-g^k\right]}{g^{(k)^T}g^k}. \end{array}$$

Or de (2.21) on aura :

$$\beta_{k+1} = \frac{g^{(k+1)^T}Ad^{(k)}}{d^{(k)^T}Ad^{(k)}} = \frac{g^{(k+1)^T}\left[g^{k+1}-g^k\right]}{g^{(k)^T}g^k},$$

ce qui démontre (2.23).

(2.24) découle alors du fait que :

$$g^{(k+1)^T}g^k = 0,$$

car

$$g^k = d^k - \beta_k d^{k-1}.$$

Appartient au sous-espace engendré par $(d^0, d^1, \cdots, d^k)$ et que g^{k+1} est orthogonal à ce sous-espace.

$c)$ Montrons enfin que d^{k+1}est conjuguée par rapport à $(d^0, d^1, \cdots, d^k)$. On a bien $d^{(k+1)^T}Ad^{(k)}$ car, en utilisant $d^{k+1} = -g^{k+1} + \beta_{k+1}d^k$ on aura :

$$\begin{array}{rcl} d^{(k+1)^T}Ad^{(k)} & = & \left(-g^{k+1}+\beta_{k+1}d^k\right)^T Ad^{(k)} \\ & = & -g^{(k+1)^T}Ad^{(k)} + \beta_{k+1}d^{(k)^T}Ad^{(k)} \\ & = & -g^{(k+1)^T}Ad^{(k)} + \dfrac{g^{(k+1)^T}Ad^{(k)}}{d^{(k)^T}Ad^{(k)}} d^{(k)^T}Ad^{(k)} \\ & = & 0. \end{array}$$

Vérifions maintenant que :

$$d^{(k+1)^T}Ad^{(i)} = 0 \text{ pour } i = 0, 1, \cdots, k-1.$$

On a :

$$d^{(k+1)^T}Ad^{(i)} = -g^{(k+1)^T}Ad^{(i)} + \beta_{k+1}d^{(k)^T}Ad^{(i)}.$$

Le seconde terme est nul par l'hypothèse de récurrence ($(d^0, d^1, \cdots, d^k)$ sont mutuellement conjuguées).

Montrons qu'il en est de même du permier terme. Puisque $x^{i+1} = x^i + \alpha_i d^i$ et que $\alpha_i \neq 0$ on a :

$$\begin{array}{rcl} Ad^i & = & \dfrac{1}{\alpha_i}(Ax^{i+1} - Ax^i) \\ & = & \dfrac{1}{\alpha_i}(g^{i+1} - g^i). \end{array}$$

En écrivant :

$$\begin{aligned} g^{i+1} &= d^{i+1} - \beta^i d^i \\ g^i &= d^i - \beta^{i-1} d^{i-1}, \end{aligned}$$

on voit que Ad^i est combinaison linéaire de d^{i+1}, d^iet de d^{i-1} seulement. Donc g^{i+1} est orthogonal au sous-espace engendré par $(d^0, d^1, \cdots, d^k)$ et comme Ad^i appartient à ce sous-espace pour $i = 0, 1, \cdots, k-1$, on en déduit $g^{(k+1)^T} Ad^{(i)} = 0$ ce qui achève la preuve. ■

Remarque 2.5 *Dans ce cas d^k est une direction de descente puisque*

$$\begin{aligned} d_k^T \nabla P(x_k) &= \left(-\nabla P(x^k) + \beta_k d^{k-1}\right)^T \nabla P(x_k) \\ &= -\nabla P(x^k)^T \nabla P(x^k) + \beta_k d^{(k-1)^T} \nabla P(x^k) \\ &= -\left\|\nabla P(x^k)\right\|^2 \quad (car\ d^{(k-1)^T} \nabla P(x^k) = 0) \\ &< 0. \end{aligned}$$

Exemple 2.2 *[9] Considérons la fonction $f : \mathbb{R}^3 \to \mathbb{R}$ définie par*

$$\min f(x_1, x_2, x_3) = \min \left\{ \frac{3}{2} x_1^2 + 2x_2^2 + \frac{3}{2} x_3^2 + x_1 x_3 + 2x_2 x_3 - 3x_1 - x_3, (x_1, x_2, x_3) \in \mathbb{R}^3 \right\}. \quad (2.25)$$

En démarrant du point initial $x^{(0)} = \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$, et en appliquant la méthode du gradient conjugué quadratique, trouver la solution optimale $\hat{x} = \begin{pmatrix} \hat{x}_1 \\ \hat{x}_2 \\ \hat{x}_3 \end{pmatrix}$ du problème (2.25)

$$\begin{aligned} f(x_1, x_2, x_3) &= \frac{1}{2}(3x_1^2 + 4x_2^2 + 3x_3^2) + x_1 x_3 + 2x_2 x_3 - 3x_1 - x_3 \\ &= \frac{1}{2}(x_1, x_2, x_3) \begin{pmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} - (x_1, x_2, x_3) \begin{pmatrix} 3 \\ 0 \\ 1 \end{pmatrix} \\ &= \frac{1}{2} x^T Q x - b^T x, \end{aligned}$$

avec

$$x^t = (x_1, x_2, x_3), \quad Q = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}, \quad b = \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix}.$$

Remarquons que la matrice $Q = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}$ *est symetrique définie positive. En effet*

$$\Delta_1 = 3 > 0, \Delta_2 = \det \begin{bmatrix} 3 & 0 \\ 0 & 4 \end{bmatrix} = 12 > 0, \Delta_3 = \det \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} = 20 > 0$$

Donc le problème (1) est un problème de minimisation quadratique. Le problème (1) admet une seule solution $\hat{x}$ *solution du système*

$$\begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix}.$$

Si on applique la méthode du gradient conjugué quadratique, $\hat{x} = x^{(3)} = \begin{bmatrix} x_1^{(3)} \\ x_2^{(3)} \\ x_3^{(3)} \end{bmatrix}$. *Calculons donc* x_3 *en partant du point initial* $x^{(0)} = \begin{bmatrix} x_1^{(0)} \\ x_2^{(0)} \\ x_3^{(0)} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$.

***Etape1** :Calcul de* $x^{(1)} = \begin{bmatrix} x_1^{(1)} \\ x_2^{(1)} \\ x_3^{(1)} \end{bmatrix}$

Calcul de g_0 *; et* d_0

On a

$$\begin{aligned} g(x) &= \nabla f(x) = Qx - b \\ &= \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} - \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 3x_1 + x_3 - 3 \\ 4x_2 + 2x_3 \\ x_1 + 2x_2 + 3x_3 - 1 \end{bmatrix} \end{aligned}$$

Donc

$$\begin{aligned} g_0 &= g\left(x^{(0)}\right) = -b = \begin{bmatrix} -3 \\ 0 \\ -1 \end{bmatrix} \\ d_0 &= -g_0 = b = \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} \end{aligned}$$

Calcul de α_0

$$\alpha_0 = -\frac{g_0^t d_0}{d_0 Q d_0} = \frac{10}{36} = 0.2778$$

Calcul de $x^{(1)}$

$$x^{(1)} = x^{(0)} + \alpha_0 d_0 = \begin{bmatrix} 0.8334 \\ 0 \\ 0.2778 \end{bmatrix}$$

***Etape2** : Calcul de* $x^{(2)} = \begin{bmatrix} x_1^{(2)} \\ x_2^{(2)} \\ x_3^{(2)} \end{bmatrix}$

Calcul de g_1,β_1 *et* d_1

$$\begin{aligned} g_1 &= \nabla f\left(x^{(1)}\right) = Qx^{(1)} - b \\ &= \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 0.8334 \\ 0 \\ 0.2778 \end{bmatrix} - \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -0.222 \\ 0.5556 \\ 0.6668 \end{bmatrix} \end{aligned}$$

$$\begin{aligned} \beta_1 &= \frac{g_1^t Q d_0}{d_0^t Q d_0} \\ &= \frac{\begin{bmatrix} -0.222 & 0.5556 & 0.6668 \end{bmatrix} \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix}}{\begin{bmatrix} 3 & 0 & 1 \end{bmatrix} \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix}} = \frac{2.892}{36} = 0.08025. \end{aligned}$$

Maintenant on est cappable de calculer d_1

$$\begin{aligned} d_1 &= -g_1 + \beta_1 d_0 \\ \begin{bmatrix} 0.222 \\ -0.5556 \\ -0.6668 \end{bmatrix} + 0.08025 \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} &= \begin{bmatrix} 0.46275 \\ -0.5556 \\ -0.58655 \end{bmatrix}. \end{aligned}$$

Calcul de α_1

$$\begin{aligned}
\alpha_1 &= -\frac{g_1^t d_1}{d_1^t Q d_1} \\
&= -\frac{\begin{bmatrix} -0.222 & 0.5556 & 0.6668 \end{bmatrix}\begin{bmatrix} 0.46275 \\ -0.5556 \\ -0.58655 \end{bmatrix}}{\begin{bmatrix} 0.46275 & -0.5556 & -0.58655 \end{bmatrix}\begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}\begin{bmatrix} 0.46275 \\ -0.5556 \\ -0.58655 \end{bmatrix}} \\
&= -\frac{-0.8025334}{3.66999753} = 0.2186740981.
\end{aligned}$$

Calcul de $x^{(2)}$

$$x^{(2)} = x^{(1)} + \alpha_1 d_1 = \begin{bmatrix} 0.8334 \\ 0 \\ 0.2778 \end{bmatrix} + 0.2186740981 \begin{bmatrix} 0.46275 \\ -0.5556 \\ -0.58655 \end{bmatrix} = \begin{bmatrix} 0.9345914389 \\ -0.1214953289 \\ 0.1495367078 \end{bmatrix}.$$

Etape3 :*Calcul de* $x^{(3)} = \begin{bmatrix} x_1^{(3)} \\ x_2^{(3)} \\ x_3^{(3)} \end{bmatrix}$

Calcul de g_2,β_2 *et* d_2

$$\begin{aligned}
g_2 &= \nabla f\left(x^{(2)}\right) = Qx^{(2)} - b \\
&= \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}\begin{bmatrix} 0.9345914389 \\ -0.1214953289 \\ 0.1495367078 \end{bmatrix} - \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -0.0466889755 \\ -0.1869079 \\ 0.1402109045 \end{bmatrix}
\end{aligned}$$

$$\begin{aligned}
\beta_2 &= \frac{g_2^t Q d_1}{d_1^t Q d_1} \\
&= \frac{\begin{bmatrix} -0.004668 & -0.186907 & 0.140210 \end{bmatrix}\begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}\begin{bmatrix} 0.46275 \\ -0.5556 \\ -0.58655 \end{bmatrix}}{\begin{bmatrix} 0.46275 & -0.5556 & -0.58655 \end{bmatrix}\begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix}\begin{bmatrix} 0.46275 \\ -0.5556 \\ -0.58655 \end{bmatrix}} = \frac{0.2595796}{3.669997} = 0.07073
\end{aligned}$$

$$d_2 = -g_2 + \beta_2 d_1 = \begin{bmatrix} 0.0466889755 \\ 0.1869079 \\ -0.1402109045 \end{bmatrix} + 0.07073 \begin{bmatrix} 0.46275 \\ -0.5556 \\ -0.58655 \end{bmatrix} = \begin{bmatrix} 0.079419283 \\ 0.147610312 \\ -0.181697586 \end{bmatrix}.$$

Calcul de α_2

$$\begin{aligned} \alpha_2 &= -\frac{g_2^t d_2}{d_2^t Q d_2} \\ &= -\frac{\begin{bmatrix} -0.046689 & -0.186908 & 0.140211 \end{bmatrix} \begin{bmatrix} 0.079419283 \\ 0.147610312 \\ -0.181697586 \end{bmatrix}}{\begin{bmatrix} 0.079419283 & 0.147610312 & -0.181697586 \end{bmatrix} \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 0.079419283 \\ 0.147610312 \\ -0.181697586 \end{bmatrix}} \\ &= -\frac{-5.677355533 \times 10^{-2}}{0.068977} = 0.8231. \end{aligned}$$

Calcul de $x^{(3)}$

$$x^{(3)} = x^{(2)} + \alpha_2 d_2 = \begin{bmatrix} 0.9345914389 \\ -0.1214953289 \\ 0.1495367078 \end{bmatrix} + 0.8231 \begin{bmatrix} 0.079419283 \\ 0.147610312 \\ -0.181697586 \end{bmatrix} = \begin{bmatrix} 0.9999614507 \\ 2.7189072 \times 10^{-6} \\ -1.85752366 \times 10^{-5} \end{bmatrix}.$$

Puisque ces calculs ont été eectués avec des arrondis, on peut supposer que la solution est

$$x^{(3)} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

En effet

$$g_3 = \nabla f\left(x^{(3)}\right) = Qx^{(3)} - b = \begin{bmatrix} 3 & 0 & 1 \\ 0 & 4 & 2 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 3 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Chapitre 3

Convergence des méthodes itératives

Nous présentons dans ce chapitre la convergence des méthodes itératives stationnaires et non stationnaires, en plus des conclusions numriéques pour les comparant les uns aux autres au sens de rapidité.

3.1 Erreur et convergence

Si x est solution de $Ax = b$ alors il vérifie

$$x = Tx + M^{-1}b. \tag{3.1}$$

Soit $e^{(k)}$ le vecteur erreur

$$e^{(k+1)} = x^{(k+2)} - x^{(k+1)} = T\left(x^{(k+1)} - x^{(k)}\right) = Te^{(k)}, \tag{3.2}$$

ce qui donne

$$e^{(k+1)} = Te^{(k)} = T^{k+1}e^{(0)}. \tag{3.3}$$

L'algorithme converge si $\lim_{k\to\infty} \left\|e^{(k)}\right\| = 0 \Leftrightarrow \lim_{k\to\infty} \left\|T^k\right\| = 0$ (c-à-d. T^k tend vers la matrice nulle).

Théorème 3.1 *(convergence de la méthode itérative)*

1· *Pour toute norme matricielle sur $M_n(\mathbb{C})$ et pour toute matrice $A \in M_n(\mathbb{C}), \rho(A) \leq \|A\|$.*

2· *Pour toute matrice $A \in M_n(\mathbb{C})$ et pour tout $\varepsilon > 0$, il existe une norme subordonnée telle que $\|A\| \leq \rho(A) + \varepsilon$, dans ce cas là la norme $\|\cdot\|$ dépend de A et de ε.*

Définition 3.1 *On dit que l'algorithme itérative (2.3) converge si pour tout $b \in \mathbb{k}^n$ et tout $x^{(0)} \in \mathbb{k}^n$, la suite itérative converge vers la solution $x = A^{-1}b$*

$$\lim_{k\longmapsto+\infty} \left\|x^{(k)} - x\right\| = 0. \tag{3.4}$$

Notant $e^{(k)} = x^{(k)} - x$ *l'erreur à la* $k-$*ème iteration, on a alors*

$$e^{(k+1)} = Te^{(k)} \Longrightarrow e^{(k)} = T^k e^{(0)}, \tag{3.5}$$

et l'algorithme converge donc si pour tout $e^{(0)}$, $\lim\limits_{k \longmapsto +\infty} T^k e^{(0)} = 0$.

Théorème 3.2 *Soit* $A \in M_n(\mathbb{k})$ *une matrice inversible.*
La méthode itérative associée à la décomposition $A = M - N$, *avec* M *inversible, converge si et seulement si le rayon spectral* $\rho(T) < 1$.

Théorème 3.3 *Soit* A *une matrice carrée. Les assertions suivantes sont équivalentes :*
1) $\lim\limits_{k \longrightarrow +\infty} A^k = 0$.
2) $\lim\limits_{k \longrightarrow +\infty} A^k v = 0$ *pour tout* v.
3) $\rho(A) < 1$.
4) $\|A\| < 1$, *pour au moins une norme matricielle induite.*

Preuve. 1)$\Longrightarrow$2)
On a $\|A\| \leq \|A^k\| \|v\|$; l'assertion 1) et la continuité de la norme implique que

$$\lim_{k \longrightarrow +\infty} \|A^k\| = 0, \tag{3.6}$$

et par suite $\lim\limits_{k \longrightarrow +\infty} \|A^k v\| = 0$ et $\lim\limits_{k \longrightarrow +\infty} A^k v = 0$ pour tout v.
2)$\Longrightarrow$3)
Ceci revient à montrer par la contraposition.
Supposons que $\rho(A) \geq 1$ et soit v le vecteur propre associé à la valeur propre λ qui vérifie $|\lambda| = \rho(A)$,
donc

$$Av = \lambda v, \tag{3.7}$$

et on en tire $A^k v = \lambda^k v$ d'où, si $|\lambda| \geq 1$ finalement $A^k v$ ne converge pas vers 0,
donc Non 2) est vraie.
3)$\Longrightarrow$4)
Pour tout $\varepsilon > 0$ il existe une norme induite telle que $\|A\| \leq \rho(A) + \varepsilon$,
il suffit de considérer un ε tel que $\rho(A) + \varepsilon < 1$, pour ce ε. Donc $\|A\| < 1$.
4)$\Longrightarrow$1)
Elle est évidente car si $\|A\| < 1$ alors $\lim\limits_{k \longrightarrow +\infty} \|A\| = 0$ et par ailleurs $\|A^k\| \leq \|A\|^k$,
d'où $\lim\limits_{k \longrightarrow +\infty} \|A^k\| = 0$ et $\lim\limits_{k \longrightarrow +\infty} A = 0$.
Ce qu'il fallait démontrer. ■

Théorème 3.4 *Soit A une matrice carrée et $\|.\|$ une norme quelconque, alors*

$$\lim_{k \longrightarrow +\infty} \left\|A^k\right\|^{\frac{1}{k}} = \rho\left(A\right) \tag{3.8}$$

Preuve. [1] i) On a $\rho\left(A\right) \leq \|A\|$ et comme $\rho\left(A^k\right) = \left(\rho\left(A\right)\right)^k$,
il s'ensuit que $\rho\left(A^k\right) = \left(\rho\left(A\right)\right)^k \leq \left\|A^k\right\|$ et par suite $\rho(A) \leq \left\|A^k\right\|^{\frac{1}{k}}$.
Donc

$$\rho(A) \leq \lim_{k \longrightarrow +\infty} \left\|A^k\right\|^{\frac{1}{k}} \tag{3.9}$$

ii) Pour montrer que $\lim\limits_{k \longrightarrow +\infty} \left\|A^k\right\|^{\frac{1}{k}} \leq \rho(A)$, introduisons pour tout $\varepsilon > 0$,
la matrice $A_\varepsilon = \frac{1}{\rho(A)+\varepsilon} A$, pour cette matrice on a $\rho(A_\varepsilon) = \frac{\rho(A)}{\rho(A)+\varepsilon} < 1$,
d'après le théorème précédent on obtient : $\|A_\varepsilon\| < 1$ pour au moins une norme matricielle induite, donc

$$\lim_{k \longrightarrow +\infty} A_\varepsilon^k = 0, \text{ et } \lim_{k \longrightarrow +\infty} \|A_\varepsilon\|^k = 0. \tag{3.10}$$

Donc $\forall \varepsilon > 0$, $\exists N\left(\varepsilon\right)$, tel que pour tout $k \geq N(\varepsilon)$ on ait : $\left\|A_\varepsilon^k\right\| < 1$ ou encore

$$\left\|A^k\right\| \leq \left(\rho\left(A\right) + \varepsilon\right)^k,$$

soit encore $\left\|A^k\right\|^{\frac{1}{k}} \leq \left(\rho\left(A\right) + \varepsilon\right)$ pour tout $\varepsilon > 0$ finalement $\lim\limits_{k \longrightarrow +\infty} \left\|A^k\right\|^{\frac{1}{k}} \leq \rho\left(A\right)$. ■

3.2 Etude de la convergence

Théorème 3.5 *Si A est une matrice carrée à diagonale strictement dominante en lignes alors la méthode de Jacobi converge.*

Preuve. On a

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}| < |a_{ii}| \qquad i = 1, \ldots, n \tag{3.11}$$

d'autre part on a : $t_{ij} = -\frac{a_{ij}}{a_{ii}}$ pour $i \neq j$ et $t_{ii} = 0$ d'où $\|T_J\|_\infty = \max\limits_i \sum\limits_j |t_{ij}| = \left\{ \frac{1}{|a_{ii}|} \sum\limits_{j \neq i} |a_{ij}| \right\}$ et d'après (1.8) on a $\|T_J\|_\infty < 1$. ■

Corollaire 3.1 *Si A est une matrice à diagonale strictemant dominante en colonnes, alors la méthode de Jacobi converge.*

Preuve. Posons $T = (D-E)^{-1}F$ et montrons que $\|T\|_\infty < 1$ où $\|T\|_\infty = \max\limits_{x\neq 0}\frac{\|Tx\|_\infty}{\|x\|_\infty}$.
Soit $y = Tx = (D-E)^{-1}Fx$ on a alors

$$(D-E)y = Fx,$$

ou encore

$$Dy = Ey + Fx \text{ et } y = D^{-1}Ey + D^{-1}Fx.$$

Considérons l'indice i_0 tel que

$$|y_{i_0}| = \max_i |y_i| = \|y\|_\infty = \|Tx\|_\infty\,. \tag{3.12}$$

Il vient :

$$y_{i_0} = \sum_{j=1}^{i_0-1}\left(D^{-1}E\right)_{i_0 j} y_j + \sum_{j=i_0+1}^{n}\left(D^{-1}F\right)_{i_0 j} x_j. \tag{3.13}$$

Par suite

$$|y_{i_0}| = \|y\|_\infty \leq \sum_{j=1}^{i_0-1}\left|\frac{a_{i_0 j}}{a_{i_0 i_0}}\right| \|y\|_\infty + \sum_{j=i_0+1}^{n}\left|\frac{a_{i_0 j}}{a_{i_0 i_0}}\right| \|x\|_\infty\,. \tag{3.14}$$

En regroupant les termes

$$\left(1 - \sum_{j=1}^{i_0-1}\left|\frac{a_{i_0 j}}{a_{i_0 i_0}}\right|\right)\frac{\|y\|_\infty}{\|x\|_\infty} \leq \sum_{j=i_0+1}^{n}\left|\frac{a_{i_0 j}}{a_{i_0 i_0}}\right|. \tag{3.15}$$

Par hypothèse, le terme $1 - \sum\limits_{j=1}^{i_0-1}\left|\frac{a_{i_0 j}}{a_{i_0 i_0}}\right|$ est strictement positif d'où on en tire :

$$\frac{\|Tx\|_\infty}{\|x\|_\infty} = \frac{\|y\|_\infty}{\|x\|_\infty} \leq \left(\sum_{j=i_0+1}^{n}\left|\frac{a_{i_0 j}}{a_{i_0 i_0}}\right|\right)\left(1 - \sum_{j=1}^{i_0-1}\left|\frac{a_{i_0 j}}{a_{i_0 i_0}}\right|\right)^{-1},$$

finalement

$$\max_{x\neq 0}\frac{\|Tx\|_\infty}{\|x\|_\infty} < 1. \tag{3.16}$$

■

Théorème 3.6 *Soit A une matrice hermitienne et inversible définie positive $(A = M - N)$ telle que M soit inversible, et la matrice $M^* + N$ soit définie positive, alors le schéma (2.3) converge si et seulement si la matrice A est définie positive.*

Preuve. i) Supposons que A est définie positive

$$A = A^* \Longrightarrow (M-N)^* = M^* - N^* = M - N.$$

D'où

$$M^{*}+N=M^{*}+M-A=M+N^{*}=(M^{*}+N)^{*}. \tag{3.17}$$

Par conséquent si A est hermitienne alors $M^{*}+N$ est aussi hermitienne.
Comme A est définie positive,
l'application $v\longrightarrow(v^{*}Av)^{\frac{1}{2}}$ définit une norme : $v\longrightarrow\|v\|_{A}=(v^{*}Av)^{\frac{1}{2}}$.
Considérons alors la norme matricielle induite par $\|.\|_{A}$ on a :

$$\left\|M^{-1}N\right\|_{A}=\left\|I-M^{-1}A\right\|_{A}=\sup_{v/\|v\|_{A}=1}\left\|\left(I-M^{-1}A\right)v\right\|_{A}. \tag{3.18}$$

En posant $\omega=M^{-1}Av$ il vient que $M\omega=Av$ et on est amené à travailler sur $\|v-\omega\|_{A}$ avec $\|v\|_{A}=1$ on a :

$$\begin{aligned}\|v-\omega\|_{A}^{2}&=(v-\omega)^{*}A(v-\omega)\\&=\|v\|_{A}^{2}-v^{*}A\omega-\omega^{*}Av+\|\omega\|_{A}^{2}\\&=1-\omega^{*}M\omega-\omega^{*}M\omega+\omega^{*}A\omega\\&=1-\omega^{*}M^{*}\omega-\omega^{*}N\omega\\&=1-\omega^{*}\left(M^{*}+N\right)\omega.\end{aligned}$$

On a $v\neq0$ donc $\omega\neq0$ et $M^{*}+N$ est définie positive donc $\omega^{*}(M^{*}+N)\omega>0$ par conséquent $\|v-\omega\|_{A}^{2}<1$ et$\|M^{-1}N\|_{A}<1$.
ii) Réciproquement, posons $T=I-M^{-1}A$ et $R=AM^{*^{-1}}(M^{*}+N)M^{-1}A$.

$$\langle Rx,x\rangle=(M+M^{*}-A)\langle y,y\rangle\ \text{ avec } y=M^{-1}Ax. \tag{3.19}$$

Or la matrice $M+M^{*}-A=M+N$ est définie positive et A est inversible, on a donc $\langle Rx,x\rangle>0,\forall x\neq0$.
Par suite R est hermitienne définie positive.
En remarquant que $A=R+T^{*}AT$, on obtient

$$\begin{aligned}A&=R+T^{*}\left(R+T^{*}AT\right)T=R+T^{*}RT+T^{*2}\left(R+T^{*}AT\right)T^{2}\\&=\sum_{i=0}^{k-1}T^{*i}RT^{i}+T^{*k}RT^{k}.\end{aligned}$$

Par hypothèse le schéma (2.3) est convergent, donc

$$\rho\left(M^{-1}N\right)<1,$$

or $M^{-1}N=I-M^{-1}A=T$ donc $\rho(T)<1$ et on a

$$\lim_{k\longrightarrow\infty}T^{k}=\lim_{k\longrightarrow\infty}T^{*k}=0.$$

Donc

$$A = \sum_{i=0}^{\infty} T^{*i} R T^{i} = R + \sum_{i=0}^{\infty} T^{*i} R T^{i}. \tag{3.20}$$

Puisque R est définie positive, et que $\langle T^{*k} R T^{k} x, x \rangle \geq 0, \forall x$, il en résulte que A est définie positive.

■

Théorème 3.7 *(Condition nécessaire de convergence).*
Si A est une matrice définie positive alors la méthode de relaxation converge si $\omega \in]0, 2[$.
Preuve. *D'après le théorème précédent, si A est définie positive et $M^* + N$ définie positive alors la méthode converge.*
Il suffit donc que $M^* + N$ soit définie positive.Or

$$M^* + N = \frac{2-\omega}{\omega} D, \tag{3.21}$$

et par suite $M^* + N$ est définie positive si $2 - \omega > 0$ c.à.d si $0 < \omega < 2$. ■

Théorème 3.8 *Le rayon spectral de la matrice de relaxation vérifie toujours l'inégalité*

$$\rho(T_\omega) \geq |\omega - 1|, \ \omega \neq 0 \tag{3.22}$$

il s'ensuit que la méthode de relaxation ne peut converger que si $\omega \in]0, 2[$.
Preuve.

$$T_\omega = \left(\frac{D}{\omega} - E \right)^{-1} \left(\frac{1-\omega}{\omega} D + F \right). \tag{3.23}$$

Si les valeurs propres de T_ω sont notées $\lambda_i(\omega)$ on a :

$$\begin{aligned} \det(T_\omega) &= \prod_{i=1}^{n} \lambda_i(\omega) \\ &= \frac{\det\left(\frac{1-\omega}{\omega} D + F\right)}{\det\left(\frac{D}{\omega} - E\right)} \\ &= (1-\omega)^n. \end{aligned}$$

D'où $\rho(T_\omega) \geq (|1-\omega|^n)^{\frac{1}{n}} = |1-\omega|$.
Pour que la méthode converge, il est nécessaire d'avoir $\rho(T_\omega) < 1$ et par conséquent $|1-\omega| < 1$ d'où $\omega \in]0, 2[$.
Ce qui d'coule la démonstration. ■

Convergence de l'Algorithme du gradient

Définition 3.2 *Un algorithme de résolution est un procède qui permet, à partir de la donnée du point initial x^0, d'engendrer la suite $x^1, x^2 ... x^k, ..,$ Un algorithme est parfaitement défini par la donnée de l'application F qui à x^k associe à $x^{k+1} = F(x^k)$: Ceci permettra de confondre un algorithme et l'application F qui lui est associe.*

Définition 3.3 *Soit $\{x^k\}_{k\in\mathbb{N}}$ une suite dans $\mathbb{R}^n$convergente vers x :*
Si

$$\lim_{k\to\infty} \sup \frac{\left\| x^{k+1} - \hat{x} \right\|}{\left\| x^k - \hat{x} \right\|^2} \prec +\infty.$$

On dit que la convergence est quadratique (superlinéaire d'ordre 2).

Convergence de l'Algorithme du gradient conjugué

Théorème 3.9 *Soit A une matrice $(n; n)$, symétrique et définie positive, le problème de minimisation sans contraintes quadratique suivant*

$$\min P(x) = \min \left\{ \frac{1}{2} x^T A x - b^T x;\ x \in \mathbb{R}^n \right\}.$$

Démarrant d'un point $x^0 \in \mathbb{R}^n$ quelconque, considérons la suite $\{x^k\}$ générée par l'algorithme du gradient conjugué quadratique définie par

$$\begin{aligned}
g^k &= \nabla P(x^k) = Ax^k - b,\ k = 0, 1, ... \\
\beta_{k+1} &= \frac{g^{(k+1)^T} A d^k}{d^{(k)^T} A d^{(k)}} \\
d^k &= \begin{cases} -g^{(0)} & \text{si } k = 0 \\ -g^{(k)} + \beta_k d^{k-1} & \text{si } k \geq 1 \end{cases} \\
\alpha_k &= \frac{-d^{(k)^T} g^k}{d^{(k)^T} A d^{(k)}},
\end{aligned}$$

et

$$x^{k+1} = x^k + \alpha_k d^k$$

Alors la suite $\{x^k\}$ converge en n itérations vers la solution optimale $\hat{x}$ du problème $P(x)$, c'est à dire que x^k vérifie $x^n = \hat{x}$ et

$$A\hat{x} = Ax^n = b$$

3.3 Tests numériques et discussions

Dans cette partie, nous présentons les expériences numériques pour faire une comparaison entre quelques méthodes itératives stationnaires et non stationnaires à l'aide d'un exemple en Matlab.

Exemple 3.1 *Soit le système $Ax = b$ telle que :*

$$\begin{pmatrix} 3 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 0 & 2 & 3 & 1 & 0 \\ 0 & 0 & 1 & 4 & 3 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

- Résoudre le système $Ax = b$ par la méthode de Jacobi, Gauss-sidel, Relaxation, Gradient et la méthode de Gradient Conjugué, avec $\omega = 1.25$, en partant de $x^{(0)} = 0$ et un tolirence 10^{-1}.

```
clear,clc
%%
%% Les variables
A=[3 1 0 0 0 ;1 2 1 0 0 ;0 2 3 1 0 ;0 0 1 4 3 ;0 0 0 1 1] ;
b=[1 ;1 ;1 ;1 ;1] ;
X0=zeros(5,1) ;tol=1e-1 ;kmax=50 ;w=1.25 ;
A=input('Donner la matrice A : ') ;
[m,n]=size(A) ;
if m==n
b=input('Donner le vecteur b : ') ;
X0=input('Donner le vecteur initial X0 : ') ;
tol=input('Donner la valeur de Tolirence : ') ;
kmax=input('Donner la nombre max d''itération : ') ;
w=input('Donner la valeur du paramètre w (0 < w < 1 sous relaxation) (1< w <2 SOR) : ') ;
D=diag(diag(A)) ;
E=triu(A,1) ;
F=tril(A,-1) ;
```

Programme de Jacobi

```
%% Jacobi
%%
```

```
Tj=D\(-E-F);
I=max(eig(Tj));
if I<1
jac_etat='Converge';
elseif I>1
jac_etat='Déverge';
else
jac_etat='Nous ne pouvons pas juger';
end
X_jac( :,1)=X0;
err_jac=1;
k_jac=0;
while err_jac>=tol && k_jac<=kmax
k_jac=k_jac+1;
for i=1 :n
S=0;
for j=1 :n
if j~=i
S=S+A(i,j)*X_jac(j,k_jac);
end
end
X_jac(i,k_jac+1)=(b(i)-S)/A(i,i);
end
err_jac=norm(X_jac( :,k_jac+1)-X_jac( :,k_jac));
end
```

Programme de Gauss Seidel

```
%% Gauss Seidel
%%
Tgs=(D+E)\(-F);
I=max(eig(Tgs));
if I<1
gs_etat='Converge';
elseif I>1
```

```
gs_etat='Déverge' ;
else
gs_etat='Nous ne pouvons pas juger' ;
end
X_gs( :,1)=X0 ;
err_gs=1 ;
k_gs=0 ;
while err_gs >= tol && k_gs<=kmax
k_gs=k_gs+1 ;
for i=1 :n
S0=0 ;
for j=1 :i-1
S0=S0+A(i,j)*X_gs(j,k_gs+1) ;
end
S1=0 ;
for j=i+1 :n
S1=S1+A(i,j)*X_gs(j,k_gs) ;
end
X_gs(i,k_gs+1)=(b(i)-S0-S1)/A(i,i) ;
end
err_gs=norm(X_gs( :,k_gs+1)-X_gs( :,k_gs)) ;
end
```

Programme de Relaxation

```
%% Relaxation
%%
M=(1/w)*D-E ;
N=((1-w)/w)*D+F ;
Tw=M\N ;
I=max(eig(Tw)) ;
if I<1
rx_etat='Converge' ;
elseif I>1
rx_etat='Déverge' ;
else
```

```
rx_etat='Nous ne pouvons pas juger';
end
X_rx( :,1)=X0;
err_rx=1;
k_rx=0;
while err_rx >= tol && k_rx<=kmax
k_rx=k_rx+1;
for i=1 :n
S0=0;
for j=1 :i-1
S0=S0+A(i,j)*X_rx(j,k_rx+1);
end
S1=0;
for j=i :n
S1=S1+A(i,j)*X_rx(j,k_rx);
end
X_rx(i,k_rx+1)=X_rx(i,k_rx)+w*(b(i)-S0-S1)/A(i,i);
end
err_rx=norm(X_rx( :,k_rx+1)-X_rx( :,k_rx));
end
```

Programme de Gradient

```
%% Gradient
g_gr( :,1)=A*X0-b;
X_gr( :,1)=X0;
d_gr( :,1)=-g_gr( :,1);
Alpha_gr(1)=(-d_gr( :,1)'*g_gr( :,1))/(d_gr( :,1)'*A*d_gr( :,1));
err_gr=1;
k_gr=1;
while err_gr >= tol && k_gr<=kmax
X_gr( :,k_gr+1)=X_gr( :,k_gr)+Alpha_gr(k_gr)*d_gr( :,k_gr);
g_gr( :,k_gr+1)=g_gr( :,k_gr)+Alpha_gr(k_gr)*A*d_gr( :,k_gr);
d_gr( :,k_gr+1)=-g_gr( :,k_gr+1);
Alpha_gr(k_gr+1)=(-d_gr( :,k_gr+1)'*g_gr( :,k_gr+1))/(d_gr( :,k_gr+1)'*A*d_gr( :,k_gr+1));
```

```
err_gr=norm(X_gr( :,k_gr+1)-X_gr( :,k_gr)) ;
k_gr=k_gr+1 ;
end
%%
```

Programme de Gradient Conjugué

```
%%
%% Gradient Conjugué
for i=1 :n
D_g(i)=det(A(1 :i,1 :i)) ;
end
if D_g>0
gc_etat='La matrice A Défini Positive' ;
else
gc_etat='La matrice A ne pas Défini Positive' ;
end
g_gc( :,1)=A*X0-b ; % g0
d_gc( :,1)=-g_gc( :,1) ; % d0
Alpha_gc(1)=-( g_gc( :,1)'*d_gc( :,1))/(d_gc( :,1)'*A*d_gc( :,1)) ; % Alpha0
X_gc( :,1)=X0 ;
err_gc=1 ;
k_gc=1 ;
while err_gc >= tol && k_gc <= kmax
X_gc( :,k_gc+1)=X_gc( :,k_gc)+Alpha_gc(k_gc)*d_gc( :,k_gc) ;
g_gc( :,k_gc+1)=A*X_gc( :,k_gc+1)-b ;
Beta(k_gc)=( g_gc( :,k_gc+1)'*A*d_gc( :,k_gc) )/(d_gc( :,k_gc)'*A*d_gc( :,k_gc)) ;
d_gc( :,k_gc+1)=-g_gc( :,k_gc+1)+Beta(k_gc)*d_gc( :,k_gc) ;
Alpha_gc(k_gc+1)=-( g_gc( :,k_gc+1)'*d_gc( :,k_gc+1))/(d_gc( :,k_gc+1)'*A*d_gc( :,k_gc+1)) ;
err_gc = norm( X_gc( :,k_gc+1)-X_gc( :,k_gc) ) ;
k_gc=k_gc+1 ;

end
```

Tableau1-Affichage des résultats1

```
Donner la matrice A = [3 1 0 0 0;1 2 1 0 0;0 2 3 1 0;0 0 1 4 3;0 0 0 1 1]
Donner le vecteur b (vecteur colonne) b = [1;1;1;1;1]
Donner le vecteur initiale (vecteur colonne) X0 = [0;0;0;0;0]
Donner la valeur de Tolirence : 10^-1
Donner la valeur du paramètre w (0 < w < 1 sous relaxation) (1< w <2 SOR): 1.25
Donner le nombre max d'itération kmax = 50
 Méthode               :  Jacobi                          Gauss Seidel                    Relaxation
 L'état                :  Converge                        Converge                        Converge
 Nombre des itération :   48                              22                              17
 La Solution           :
    X(1) =                 0.686476                0.655813                        0.697137
    X(2) =                 -1.09177                -0.995026                       -1.11233
    X(3) =                 2.46959                 2.37483                         2.55513
    X(4) =                 -4.36799                -4.19454                        -4.47244
    X(5) =                 5.28992                 5.19454                         5.48299
    Erreur :               9.495542e-002               9.867150e-002                       9.006124e-002


 Méthode               :      Gradient                            Gradient Conjugué
 Nombre des itération :       36                                  16
 La Solution           :
    X(1) =                0.558906                        0.806067
    X(2) =                -0.680791                       -1.35944
    X(3) =                1.88445                         2.90176
    X(4) =                -3.28212                        -4.96324
    X(5) =                4.1728                          5.98713
 Erreur :                 9.524890e-002                   8.161530e-002
 Alpha  :                 0.439879                    0.38056
 Beta   :                                                 -0.000465524
 L'état de A   :                                          La matrice A Défini Positive
```

Affichage des résultats

Exemple 3.2 *Appliquer la Méthode du Gradient et Gradient Conjugué dans le cas quadratique au probléme suivant :*

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

où

$$Q = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 1 & 2 & 3 & 3 \\ 1 & 2 & 3 & 4 \end{pmatrix}, b = \begin{pmatrix} -4 \\ -7 \\ -9 \\ -10 \end{pmatrix}$$

avec un vecteur initiale $x^{(0)} = 0$ *et un tolirence* 10^{-3}.

Nous faisons le même algorithme antérieur, nous obtenons donc les résultats suivants :

Tableau2-Affichage des résultats2

```
Donner la matrice A = [1 1 1 1;1 2 2 2;1 2 3 3;1 2 3 4]
Donner le vecteur b (vecteur colonne) b = [-4;-7;-9;-10]
Donner le vecteur initiale (vecteur colonne) X0 = [0;0;0;0]
Donner la valeur de Tolirence : 10^-3
Donner le nombre max d'itération kmax = 50
     Méthode               :          Gradient                Gradient Conjugué
     Nombre des itération :          26                           6
     La Solution           :
                   X(1) =        -0.0623842                   -1
                   X(2) =        -0.712531                -1
                   X(3) =        -1.2487                  -1
                   X(4) =        -1.54177                 -1
                   Erreur :  9.818678e-004                1.113942e-013
                   Alpha  :  0.554618                 1.18152
                   Beta   :                                 -0.00104368
           L'état de A    :                       La matrice A Défini Positive
```

Tableau3-Ladirection d^k

```
>> d_gr

d_gr =

  Columns 1 through 16

   -4.0000    0.3770   -0.0796    0.0371   -0.0604    0.0292   -0.0453    0.0224   -0.0338    0.0169   -0.0252    0.0127   -0.0188    0.0095   -0.0141    0.0071
   -7.0000    0.2371   -0.0268   -0.0465   -0.0202   -0.0334   -0.0147   -0.0246   -0.0105   -0.0185   -0.0074   -0.0141   -0.0052   -0.0108   -0.0036   -0.0083
   -9.0000   -0.0574   -0.0612   -0.0391   -0.0458   -0.0272   -0.0347   -0.0188   -0.0265   -0.0129   -0.0204   -0.0088   -0.0158   -0.0060   -0.0123   -0.0040
  -10.0000   -0.2651   -0.1239    0.0055   -0.0874    0.0017   -0.0627    0.0000   -0.0454   -0.0007   -0.0330   -0.0011   -0.0242   -0.0012   -0.0178   -0.0011

  Columns 17 through 26

   -0.0106    0.0053   -0.0080    0.0040   -0.0060    0.0030   -0.0046    0.0023   -0.0035    0.0017
   -0.0026   -0.0064   -0.0018   -0.0050   -0.0013   -0.0039   -0.0009   -0.0030   -0.0007   -0.0024
   -0.0097   -0.0026   -0.0076   -0.0017   -0.0060   -0.0011   -0.0048   -0.0007   -0.0039   -0.0004
   -0.0132   -0.0011   -0.0098   -0.0010   -0.0073   -0.0009   -0.0055   -0.0008   -0.0042   -0.0007

>> d_gc

d_gc =

   -4.0000   -0.3814   -0.0494   -0.0021   -0.0000   -0.0000
   -7.0000   -0.2449    0.0346    0.0046   -0.0000   -0.0000
   -9.0000    0.0475    0.0239   -0.0055   -0.0000    0.0000
  -10.0000    0.2540   -0.0264    0.0026   -0.0000    0.0000
```

Tableau4-Legradient g^k

```
>> g_gr

g_gr =

  Columns 1 through 16

    4.0000   -0.3770    0.0796   -0.0371    0.0604   -0.0292    0.0453   -0.0224    0.0338   -0.0169    0.0252   -0.0127    0.0188   -0.0095    0.0141   -0.0071
    7.0000   -0.2371    0.0268    0.0465    0.0202    0.0334    0.0147    0.0246    0.0105    0.0185    0.0074    0.0141    0.0052    0.0108    0.0036    0.0083
    9.0000    0.0574    0.0612    0.0391    0.0458    0.0272    0.0347    0.0188    0.0265    0.0129    0.0204    0.0088    0.0158    0.0060    0.0123    0.0040
   10.0000    0.2651    0.1239   -0.0055    0.0874   -0.0017    0.0627   -0.0000    0.0454    0.0007    0.0330    0.0011    0.0242    0.0012    0.0178    0.0011

  Columns 17 through 26

    0.0106   -0.0053    0.0080   -0.0040    0.0060   -0.0030    0.0046   -0.0023    0.0035   -0.0017
    0.0026    0.0064    0.0018    0.0050    0.0013    0.0039    0.0009    0.0030    0.0007    0.0024
    0.0097    0.0026    0.0076    0.0017    0.0060    0.0011    0.0048    0.0007    0.0039    0.0004
    0.0132    0.0011    0.0098    0.0010    0.0073    0.0009    0.0055    0.0008    0.0042    0.0007

>> g_gc

g_gc =

    4.0000    0.3770    0.0427    0.0014    0.0000    0.0000
    7.0000    0.2371   -0.0389   -0.0041    0.0000    0.0000
    9.0000   -0.0574   -0.0230    0.0058    0.0000         0
   10.0000   -0.2651    0.0309   -0.0029    0.0000         0
```

Tableau5-Lepas α_k et lescalaire β_k

```
>> Alpha_gr

Alpha_gr =

  Columns 1 through 16

    0.1208    1.0199    0.1457    0.5418    0.1458    0.5432    0.1459    0.5442    0.1460    0.5451    0.1460    0.5461    0.1461    0.5471    0.1462    0.5483

  Columns 17 through 26

    0.1463    0.5496    0.1464    0.5508    0.1465    0.5521    0.1466    0.5534    0.1466    0.5546

>> Alpha_gc

Alpha_gc =

    0.1208    1.0295    2.3717    3.3912    0.1206    1.1815

>> Beta

Beta =

    0.0011    0.0177    0.0126    0.0000   -0.0010
```

Tableau6-Lasolution x^k

```
>> X_gr

X_gr =

  Columns 1 through 16

        0   -0.4831   -0.0985   -0.1101   -0.0900   -0.0988   -0.0829   -0.0895   -0.0774   -0.0823   -0.0731   -0.0768   -0.0699   -0.0726   -0.0675   -0.0695
        0   -0.8454   -0.6035   -0.6074   -0.6326   -0.6356   -0.6537   -0.6559   -0.6693   -0.6708   -0.6809   -0.6820   -0.6896   -0.6904   -0.6963   -0.6968
        0   -1.0869   -1.1455   -1.1544   -1.1756   -1.1823   -1.1970   -1.2021   -1.2123   -1.2162   -1.2233   -1.2262   -1.2311   -1.2334   -1.2367   -1.2385
        0   -1.2077   -1.4780   -1.4961   -1.4931   -1.5058   -1.5049   -1.5140   -1.5140   -1.5207   -1.5211   -1.5259   -1.5265   -1.5300   -1.5306   -1.5332

  Columns 17 through 26

  -0.0656   -0.0672   -0.0643   -0.0654   -0.0632   -0.0641   -0.0625   -0.0631   -0.0619   -0.0624
  -0.7014   -0.7017   -0.7053   -0.7055   -0.7083   -0.7085   -0.7106   -0.7108   -0.7124   -0.7125
  -1.2407   -1.2421   -1.2435   -1.2446   -1.2456   -1.2465   -1.2471   -1.2478   -1.2481   -1.2487
  -1.5339   -1.5358   -1.5364   -1.5378   -1.5384   -1.5394   -1.5399   -1.5407   -1.5412   -1.5418

>> X_gc

X_gc =

        0   -0.4831   -0.8758   -0.9930   -1.0000   -1.0000
        0   -0.8454   -1.0974   -1.0155   -1.0000   -1.0000
        0   -1.0869   -1.0380   -0.9814   -1.0000   -1.0000
        0   -1.2077   -0.9461   -1.0087   -1.0000   -1.0000
```

Chapitre 4

Conclusion

Dans ce mémoire, nous avons discuté de quelques méthodes itératives importantes de résolution d'un système linéaire .

La méthode itérative contrairement à d'autres à l'avantage de ne pas avoir besoin de garder en mémoire la totalité d'un matrice de très grande taille gourmande en capacités mémoire. Cette méthode permet de garder en mémoire que les coefficient non nuls d'une matrice de grande taille. Cependant, le succès de calcul n'est pas assuré quelque soit la matrice, certaines conditions sont nécessaire afin d'obtenir un résultat convergent.

Pour toute méthode itérative, il convient de s'assurer que la convergence est suffisamment rapide pour que le temps de calcul ne soit pas consommé sans que la recherche d'une solution ne soit réellement effectuée.

Ou nous avons comparé entre la méthodes itératives stationnaire et non stationnaire en termes de rapidité.

On remarque que la méthode de Jacobi est très lente, c'est pour cette raison on a l'amilioré par la méthode de Gauss-Seidel.

Dans la méthode de Gauss-Seidel , si le rayon spectrale est très proche de 1, alors la convergence devient lente. On a done introduit le paramètre ω qui est accéléré la convergence (i.e le nombre d'itération diminuer).

En revanche dans les méthodes non stationnaires, la méthode du gradient conjugué est plus efficace que la méthode du gradient.

Il s'agit d'inclure un calcul de valeur β_k qui permet d'accéder à la solution avec des itérations minimales.

Le seul inconvénient dans cette méthode est quand la taille de la matrice est trés grand. Ce qui nous conduit a un mal conditionnement. Si la matrice A est mal conditionnée, la convergence de l'algorithme du gradient conjugué est lente. Dans ce cas, on recoure à améliorer la vitesse de convergence. Le problème est dans le choix de la matrice de pré conditionnement.

Comment choisi un pré conditionné est une question fort délicate qui à ce jour n'a pas de solution universelle. En analyse numérique.

Bibliographie

[1] B. Abdesslam et D. Mohamed, Polycopie Analyse Numérique. Section SMA-SMI/S2. Université Mohamed 1erFaculté Des Sciences D'udjda-Maroc. 2010-2011.

[2] A. Bjoreck, Numerical methods in matrix computations. Springer International Publishing Switzerland, 2015.

[3] R. Burden and D. Faires, Numerical analysis. unge-Kutta,Université de Grenoble, Master.

[4] P. G. Ciarlet, Introduction à l'analyse numérique matricielle et à l'optimisation. Dunod 1998, Masson 1990.

[5] W. Ford, Numerical lineare algebra with applications. Ac.Press 2010.

[6] F. Jedrzejewski, Introduction aux méthodes numériques, Deuxième édition. Springer-verlage france, paris 2005.

[7] Y. Liu and C. Storey, Efficient generalized conjugate gradient algorithms, Part 1 - Theory, Journal of Optimization Theory and Applications, Vol. 69 (1991), 129-137.

[8] G. Liu, J. Han and H. Yin, Global convergence of the Fletcher-Reeves algorithm with inexact line search, Appl. Math. J. Chinese Univ. Ser. B. 10 (1995) 75-82.

[9] B. Rachid, Optimisation sans contraintes. Tome2. Méthodes du gradient conjugue, (2016) 19-23.

[10] J. Rappaz et M. Picasso, Introduction à l'analyse numérique. Presses polytechnique et univesitaires romandes, 2004.

[11] R. Sacco et F. Saleri, méthodes numériques, Springer-Verlag Italia, Milano 2007.

[12] R. Sacco et P. Gervasio, Calcul Scientifique-Deuxième édition, Springer-Verlag Italia 2010.

[13] M. Schatzman, Analyse numérique. Cours et exercices pour la licence. InterEditions 1991.

[14] D. Serre, Les matrices. Dunod, Masson Sciences 2001.

[15] H. Soulami, Méthodes de descente en optimisation numérique. Mémoire de fin d'études de licences. Université Sidi Mohamed Ben Abdellah, (2016).