



الجمهورية الجزائرية الديمقراطية الشعبية
REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
وزارة التعليم العالي و البحث العلمي
ISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE
جامعة عباس لغرور خنشلة
UNIVERSITE ABBES LAGHROUR- KHENCHELA



Faculté des Sciences et de la Technologie
Département : Mathématiques & Informatique

N° de série :.....

Mémoire de fin d'études
Pour l'obtention du diplôme de Master

Filière: Mathématiques
Spécialité: Mathématiques Appliquées

Sur l'Optimisation des Systèmes Creux

Réalisé par : LAOUAR Chokri

Dirigé par : SAOUDI Khaled
MCA Université de Khenchela

Membre de jury :

Président : GUEMMAZ Abderrahim MAA Université de Khenchela
Examineur : BENHADID Ayache MAA Université de Khenchela

Présenté le : 13/09/2020

- Session 2019/2020 -

Résumé :

Le fait que la matrice d'un système d'équation linéaire creux (01) est aussi creuse ! Il est intéressant de ne pas stocker les zéros, ceci permet, d'une part un gain de place, d'autre part un gain de temps afin d'obtenir une solution optimale [10].

Le but de ce travail est consacré à une étude comparative des méthodes itératives pour résoudre (01), où on a revendiqué particulièrement l'application de la méthode itérative de Relaxation et une méthode d'optimisation, représentée par des méthodes de minimisations, tel que l'essai de celle du gradient est avérée excellente [3] !

Abstract:

The fact that the matrix of a system of sparse linear equations (01) is also sparse! It is advantageous not to store the zeros, this allows, on the one hand a saving of space, on the other hand a saving of time in order to obtain an optimal solution [10].

The aim of this work is devoted to a comparative study of iterative methods to solve (01), where we notably claimed the application of the iterative method of Relaxation and of an optimization method, represented by minimization methods, such as the gradient test turned out to be excellent [3]!

ملخص:

بما أن مصفوفة النظام الخطي الجوفي (01) جوفية كذلك! من المستحسن عدم تخزين الأصفار، فهذا يتيح من ناحية توفير المساحة، ومن ناحية أخرى توفير الوقت من أجل الحصول على الحل الأمثل [10]!
الهدف من هذا العمل هو دراسة و مقارنة بين الطرق الناجعة لحل النظام الخطي الجوفي (01)، حيث انتهجنا بشكل خاص تطبيق الطريقة التقريبية للاسترخاء وطريقة التحسين، الممثلة بطرق التصغير، أين أثبتت استعمال طريقة التدرج أنها ممتازة [3]!

Remerciements

Mes remerciements s'adressent à : M. Khaled Saoudi pour l'attention et la disponibilité dont il a su faire preuve au cours de son encadrement.

Je voudrai également remercier les membres de mon jury : M. Benhahid Ayache et M. Guemmaz Abderrahim pour l'honneur qu'ils m'ont fait en portant leur attention sur ce travail.

Dédicaces

Tous les mots ne sauraient exprimer la gratitude, l'amour, le respect et la reconnaissance, c'est tout simplement : Je dédie ce mémoire de Master à :

Ma tendre Mère : Tu représentes pour moi la source de tendresse et l'exemple de dévouement qui n'a pas cessé de m'encourager.

Tu as fait plus qu'une mère puisse faire pour que ses enfants suivent le bon chemin dans leur vie et leurs études.

À Mon très cher Père : Aucune dédicace ne saurait exprimer l'amour, l'estime, le dévouement et le respect que j'ai toujours pour vous.

Rien au monde ne vaut les efforts fournis jour et nuit pour mon éducation et mon bien être.

Ce travail et le fruit de tes sacrifices que tu as consentis pour mon éducation et ma formation le long de ces années....

Table des matières

Introduction :	7
-----------------------------	----------

Chapitre 01

Systeme linéaire creux :	9
Généralités sur l'optimisation :	9
Existence de solution :	10
Classification des Méthodes :	11
Méthodes Directes :	11
Méthodes Indirectes (Itérative) :	12
Conditionnement d'un Systeme Linéaire :	12
Structure d'espace vectoriel :	13
Applications linéaires :	15
Schéma de Stockage Icompressé :	16
Schéma de Stockage Compressé :	16

Chapitre 02

Méthode de Relaxation :	20
Erreurs de la méthode :	23
Algorithme :	23
Convergence de la Relaxation :	25
Condition de convergence :	26

Comparaisons : 27

Chapitre 03

Méthode de descente (Gradient) :30

Convergence :33

Exemple d'Application :35

Conclusion : 38

Bibliographies : 39

Introduction

L'étude générale des systèmes d'équations linéaires est relativement fondamentale d'un point de vue mathématique ! Formellement en raison de la linéarité de ces équations, on peut mettre en œuvre deux types de méthodes, les méthodes directes qui consistent à manipuler directement leurs équations pour achever la solution et les méthodes indirectes, appelées aussi itératives, destinées aux systèmes de grande dimension ou mal conditionnés, tels que les systèmes creux, problématique de notre thème, où leur résolution est un outil essentiel pour la simulation de phénomènes physiques dans différents domaines (physique des matériaux, dynamique des fluides, ...etc.).

Le mémoire est organisé comme suit :

Dans la première partie du premier chapitre sont donnés à titre de rappel, quelques résultats préliminaires, particulièrement d'algèbre et d'analyse pour entamer les chapitres suivants et dans la seconde partie, nous citons la procédure du stockage des matrices creuse !

Dans le second chapitre, nous considérons une étude globale pour résoudre le système linéaire creux (01) et pour arriver à une solution optimale, on a revendiqué particulièrement l'application de la méthode itérative de Relaxation.

Les conséquences du deuxième chapitre ont volontairement introduits une Idée principale pour entamer ce troisième chapitre (centre du mémoire), qui juge l'existence d'un tel isomorphisme entre deux méthodes, signalé par des critiques sur la convergence de la Relaxation, qui ont transformées notre problème en un problème d'optimisation, représenté par des méthodes de minimisations, tel que l'essai de celle du gradient est avérée excellente.

À titre de conclusion, on donne certaines remarques importantes relatives aux méthodes étudiées et à quelques développements possibles.

Enfin, il convient de signaler que notre travail est essentiellement basé sur une combinaison de méthodes déjà vues (étudiées).

Chapitre 1

Préliminaire

Le but de ce chapitre est de rappeler les notions et résultats classiques fondamentaux de l'algèbre et de l'analyse numérique utiles pour aborder les deux chapitres suivants, citer la procédure du stockage des matrices creuses et on pourra aussi, consulter une abondante bibliographie citée dans les références.

Première partie:

1. Position de Problème :

On s'intéresse à calculer la solution $X \in \mathbb{R}^n$ d'un système d'équations linéaires, dont la matrice $A : M_{n \times n}(\mathbb{R}) \rightarrow M_{n \times n}(\mathbb{R})$ est creuse.

1.1 Système linéaire creux

Soit (01) le système d'équations linéaires creux :

$$A X = B \quad (01)$$

Où $B \in \mathbb{R}^n$.

1.2 Matrices Creuses :

1.2.1. Définition (01)

Dans la discipline de l'analyse numérique des mathématiques, une matrice creuse est une matrice contenant beaucoup de zéros.

Le fait que la matrice du système linéaire (01) est creuse, il est intéressant de ne pas stocker les zéros. Ceci permet, d'une part un gain de place, d'autre part un gain de temps lors du calcul du produit de la matrice par un vecteur, une étude de stockage est faite pour éviter de classer les valeurs non significatives (zéros).

2) Généralités sur l'optimisation

2.1) Définition (02)

L'optimisation est une branche des mathématiques cherchant à modéliser, à analyser et à résoudre analytiquement ou numériquement les problèmes qui consistent à minimiser ou maximiser une fonction sur un ensemble.

Analyse et formulation du problème :

On peut représenter un polyèdre convexe de différentes manières. Lorsqu'on le voit comme ci-dessus, à savoir comme une intersection d'un nombre fini de demi-espaces :

$$\{x \in \mathbb{R}^n : Ax \leq b\}$$

on dit que le polyèdre (ou le problème d'optimisation linéaire avec un tel polyèdre) est écrit sous forme canonique. Lorsque le polyèdre convexe est vu comme l'intersection de l'orthant positif et d'un sous-espace affine

$$\{x \in \mathbb{R}^n : Ax = b, x \geq 0\}$$

on dit que le polyèdre (ou le problème d'optimisation linéaire avec un tel polyèdre) est écrit sous forme standard.

En pratique, les problèmes peuvent présenter des contraintes plus variées ou plus structurées telles que des *contraintes d'égalité* ou des *contraintes de borne* inférieures et supérieures

$$A_E x = b, \quad l_I \leq A_I x \leq u_I, \quad l_B \leq x \leq u_B$$

Les analyses du problème d'optimisation linéaire se font le plus souvent sur le problème dont l'ensemble admissible est représenté sous la forme standard, problème que nous écrirons comme suit :

$$(P_L) \quad \begin{cases} \inf_x C^T x \\ Ax = b \\ x \geq 0. \end{cases}$$

On note

$$\text{Val}(P_L) = \inf_x C^T x \text{ tel que : } Ax=b, x \geq 0$$

la valeur optimale de (P_L) et

$$S_P = S(P_L) := \{x \in \mathbb{R}^n : Ax = b, x \geq 0, C^T x = \text{val}(P_L)\}$$

2.2) Existence de solution :

Il y a exactement deux cas (exclusifs) dans lesquels le problème d'optimisation linéaire n'a pas de solution.

Le premier est celui où les contraintes ne sont pas compatibles (par exemple $x \geq 2$ et $x \leq 1$). La valeur optimale du problème de minimisation vaut alors $+\infty$. Par convention. Dans ce cas, on dit que le problème n'est pas réalisable.

Le second se produit lorsque le problème de minimisation est réalisable mais que sa valeur optimale vaut $-\infty$ (par exemple lorsqu'on cherche à minimiser x sous la contrainte $x \leq 0$). Dans ce cas, on dit que le problème n'est pas borné ou est *non borné*.

2.3) Domaine d'application de l'optimisation

L'optimisation est essentiellement appliquée pour résoudre des problèmes d'optimisation à moyen et long terme (problèmes stratégiques et tactiques, dans le vocabulaire de la recherche opérationnelle). Les domaines d'application de ces problèmes sont très nombreux aussi bien dans la nature des problèmes abordés (planification et contrôle de la production, distribution dans des réseaux) que dans les secteurs d'industrie :

Industrie manufacturière, énergie (pétrole, gaz, électricité, nucléaire), transports (aériens, routiers et ferroviaires), télécommunications, industrie forestière, finance

3) Classification des Méthodes

Considérons un large système d'équations linéaires creux de la forme (01) où A est une matrice d'ordre $(N \times N)$, B et X deux vecteurs de taille N .

Il existe deux méthodes de résolution :

3.1 Méthodes Directes

Les méthodes de résolution directe procèdent d'abord à une transformation du système original, puis à la résolution du système transformé. Cette transformation consiste à factoriser la matrice, puis la résolution s'appliquera à des systèmes triviaux (faciles), tels que les systèmes triangulaires, diagonales, ..., de ce fait, une méthode directe conduit à la solution exacte, après un nombre d'opérations élémentaires connu et fini.

Parmi ces méthodes on peut citer l'élimination de Gauss, la factorisation QR, ..., etc. En pratique, elles sont utilisées lorsque $\text{Rg}(A) \leq 10^5$.

3.2 Méthodes Indirectes (Itératives)

Les méthodes itératives sont souvent utilisées pour résoudre les systèmes (01) de grande dimension dont la matrice A est très creuse.

Elles engendrent une suite d'approximations de la solution, donnant une réponse approximative en un nombre infini d'opérations.

Dans la pratique, l'approximation est souvent satisfaisante au bout d'un nombre d'itérations relativement petit.

Elles ont une grande stabilité relativement aux erreurs d'arrondis.

Parmi ces méthodes, on peut distinguer la méthode de Jacobi, Gauss-Seidel, Relaxation, ..., etc.

3.3) Conditionnement d'un Système Linéaire

En dimension finie, toutes les normes sont équivalentes, en pratique on choisit celle qui nous arrange.

3.3.1) Définition (03)

Soit $A = (a_{ij})_{N \times N} \in M_n(\mathbb{R})$, la norme de maximum de la matrice A est définie par :

$$\|A\|_{\infty} = \max_{1 \leq j \leq N} \sum_{i=1}^n |a_{ij}|$$

3.3.2) Définition (04)

Le rayon spectral ρ de la matrice A est défini par :

$$\rho(A) = \max\{|\lambda| \mid \text{tel que } \lambda \text{ est une valeur propre de } A\}$$

3.3.3) Lemme (01)

Pour toute matrice A de type $(N \times N)$ et tout $\varepsilon > 0$, il existe au moins une norme $\|A\|$, telle que :

$$\rho(A) \leq \|A\| \leq \rho(A) + \varepsilon$$

3.3.4) Théorème (01)

Soit : $B \in \mathbb{R}^n$ et $A = M - N \in M_n(\mathbb{R})$ non singulière. Si M est non singulière et si : $\rho(M^{-1}N) < 1$
alors les itérations $X^{(k)}$ définies par : $MX^{(k+1)} = NX^{(k)} + B$
convergent vers $X = A^{-1}B$, quelque soit la valeur initiale $X^{(0)}$.

3.4) Convergence

3.4.1) Définition (05)

Une matrice $A \in M_n(\mathbb{R})$ est dite à diagonale dominante stricte si :

$$|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}| \quad \text{Pour : } i = \overline{1, n} .$$

4) Structure d'espace vectoriel

4.1) Définition (06)

L'espace vectoriel réel \mathbb{R}^n est le prototype d'espace vectoriel réel de dimension finie. Ses éléments sont les n-uplets ordonnés de nombres réels (x_1, x_2, \dots, x_n) , que nous écrirons sous la forme de vecteur colonne :

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix}$$

Les opérations d'addition et de multiplication de nombre réels permettent de définir deux opérations sur \mathbb{R}^n , l'addition de vecteurs :

$$\begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix} + \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 + y_1 \\ x_2 + y_2 \\ \cdot \\ \cdot \\ x_n + y_n \end{bmatrix}$$

et la multiplication d'un vecteur par un nombre réel, pour tout réel λ ,

$$\lambda \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{bmatrix} = \begin{bmatrix} \lambda x_1 \\ \lambda x_2 \\ \cdot \\ \cdot \\ \lambda x_n \end{bmatrix}$$

Ces deux opérations ont des propriétés héritées de celles satisfaites par les opérations d'addition et de multiplication sur les nombres réels. On peut vérifier ainsi que, l'addition de n-uplets satisfait les relations suivantes :

- i) pour tous $x, y, z \in \mathbb{R}^n$, on a $(x+y) + z = x + (y+z)$,
- ii) pour tous $x, y \in \mathbb{R}^n$, on a $x+y = y+x$,
- iii) si 0 désigne le n-uplet $(0,0,\dots,0)$, on a $x+0 = x$,
- iv) en notant $-(x_1, x_2, \dots, x_n)$, le n-uplet $(-x_1, -x_2, \dots, -x_n)$, on a $x + (-x) = 0$.

D'autre part, la multiplication par un nombre réel satisfait les relations suivantes :

- v) pour tous $\lambda, \mu \in \mathbb{R}$ et tout $x \in \mathbb{R}^n$, on a $\lambda(\mu x) = (\lambda \mu)x$,
- vi) pour tout $x \in \mathbb{R}^n$, on a $1x = x$.

Enfin, les opérations d'addition et de multiplication par un réel vérifient des relations de compatibilités entre elles :

- vii) pour tout $\lambda \in \mathbb{R}$ et tous $x, y \in \mathbb{R}^n$, on a $\lambda(x+y) = \lambda x + \lambda y$.
- viii) pour tous $\lambda, \mu \in \mathbb{R}$ et tout $x \in \mathbb{R}^n$, on a $(\lambda + \mu)x = \lambda x + \mu x$.

4.1.1) Combinaisons linéaires

Soit E un K -espace vectoriel. Soient x_1, x_2, \dots, x_p des vecteurs de E . On appelle combinaison linéaire des vecteurs x_1, x_2, \dots, x_p tout vecteur de la forme :

$$\lambda_1 x_1 + \lambda_2 x_2 + \dots + \lambda_p x_p$$

où $\lambda_1, \dots, \lambda_p \in K$

Plus généralement, si $(x_i)_{i \in I}$ est une famille de vecteurs de E , on appelle combinaison linéaire des $(x_i)_{i \in I}$ toute somme

$$\sum_{i \in I} \lambda_i x_i$$

dans laquelle, pour tout $i \in I$, $\lambda_i \in K$ et où les λ_i sont tous nuls, sauf un nombre fini.

4.2) Applications linéaires

4.2.1) Définition (07)

Soient E et E' deux K -espaces vectoriels. Une application

$$u : E \rightarrow E'$$

est dite linéaire si, pour tous vecteurs $x, y \in E$ et scalaires $\lambda, \mu \in K$,

$$u(\lambda x + \mu y) = \lambda u(x) + \mu u(y).$$

Il découle de cette définition que $u(0) = 0$.

4.2.2) Espace vectoriel des applications linéaires

— On notera $L(E, E')$ l'ensemble des applications linéaires de E dans E' .

Si u et v sont deux applications linéaires de E dans E' et $\lambda \in K$ un scalaire, Alors les applications $u+v$ et λu définies, pour tout $x \in E$, par :

$$(u+v)(x) = u(x) + v(x),$$

$$(\lambda u)(x) = \lambda u(x),$$

sont aussi des applications linéaires de E dans E' . L'addition et la multiplication par un scalaire sur l'espace vectoriel E' induisent ainsi une structure de K -espace vectoriel sur $L(E, E')$.

Une application linéaire de E dans E est appelée un endomorphisme de E . On notera $L(E)$ l'espace vectoriel des endomorphismes de E .

Une application linéaire de E dans K est appelée une forme linéaire. Une application linéaire et bijective est appelée un isomorphisme. Les endomorphismes bijectifs de E sont encore appelés les automorphismes de E . Les automorphismes de E , muni de la composition, forment un groupe, noté $GL(E)$ et appelé le groupe linéaire de E .

Deuxième partie:

Dans cette deuxième partie, nous présentons quelques formats de stockage permettant d'économiser la mémoire et de faciliter les parallélises pour les matrices creuses de très grandes tailles.

Problématique

Le volume important de donnée cause des problèmes de stockages et de temps de calcul, donc les applications relatives sont parallélisées pour la haute performance.

Notre but est la manipulation d'une matrice creuse, en évitant de stocker les valeurs non significatifs (nnz).

Plusieurs formats existent, chacun est adapté à une classe de matrice, parmi ces formats citons : schéma de stockage compressé, schéma de stockage incompressé, ..., etc.

1- Schéma de Stockage Icompressé :

La version donnée par ce schéma permet de stocker, les non zéros de la matrice L et U, ligne par ligne, les valeurs significatives sont stockées en fonction de l'accroissement de l'indice des colonnes.

Pour identifier les valeurs de chaque ligne, il est nécessaire de connaître, le début de la ligne, le nombre des valeurs qu'elle contient et les indices des colonnes correspondantes.

Le stockage de la matrice A (matrice initiale), exige trois sortes de tableaux (*IA, JA et A*) :

Le tableau *A*: contient les valeurs non nulles de A, stockées ligne par ligne. Le tableau IA : contient (N+1) pointeurs qui délimitent les lignes de *A*.

Le tableau *JA*: possède une entrée pour chaque entrée de *A*, le nombre stocké est approximativement égale aux nombre des nnz de la matrice A.

2- Schéma de Stockage Ccompressé :

Il présente un nombre d'opération supérieure à celui constaté dans le schéma précédent.

Cependant les exigences de stockage en espace sont pratiquement réduites.

Ce schéma permet de stocker les éléments de la matrice triangulaire L par colonne (ou L^t est stockée par ligne).

On constate que ce type de schéma, comme son nom l'indique, permet une certaine compression des tableaux *A, JA et IA*, même ceux correspondant aux matrices *L et U*, appelés : *JL, IL, JU et IU*.

Il paraît évident, que les éléments diagonaux des matrices triangulaires *L et U*, qui sont égaux à 1 par hypothèse, ne seront pas stockés.

La caractéristique remarquable des valeurs unités est représentée par les premiers éléments de chaque ligne.

Afin d'éviter le stockage de la totalité des éléments nuls, les matrices creuses sont stockées sous des formats comprimés différents de celui des matrices denses.

Chapitre 2

RÉSOLUTION DE SYSTEMES LINEAIRES CREUX

L'objectif de ce chapitre est d'étudier (01) par des méthodes itératives, dont on a préféré la "Relaxation" qui peut être considérée comme une bonne méthode numérique, aussi une réflexion secondée par un passage propre aux méthodes d'optimisations qui seront bien développées dans le troisième chapitre.

1) Méthode Indirecte

Sont aussi appelées méthodes itératives, destinées aux systèmes de grande dimension, mal conditionnés,...etc, tel que les systèmes creux où leurs solutions est sollicitées par le biais d'une convergence d'un processus itératif.

Elle consiste d'abord à passer au système $X = TX + V$, et sa solution est donnée comme une limite de la suite définie par:

$X_K = TX_{K-1} + V$, où X_0 étant une approximation initiale, en pratique, elles sont utilisées lorsque le $\text{rang}(A) \geq 10^5$.

Le principe général de ces méthodes consiste à résoudre le système linéaire :

$$AX = B \quad (02)$$

Tout en commençant par le développement de la matrice A sous la forme :

$A = M - N$ où M doit être régulière, alors (02) peut s'écrire :

$$(M - N)X = B$$

ou encore : $MX = NX + B$

La méthode itérative associée à l'égalité précédente, consiste à partir d'un choix de vecteur initial x_0 a générée la suite x_1, x_2, \dots, x_n de la manière suivante :

$$\begin{cases} x_1 = M^{-1}Nx_0 + M^{-1}b_1 \\ x_2 = M^{-1}Nx_1 + M^{-1}b_2 \\ \dots\dots\dots \\ x_{k+1} = M^{-1}Nx_k + M^{-1}b_n \end{cases}$$

Cette suite d'égalités peut être représentée par la relation formelle itérative suivante:

$$X_{k+1} = TX_k + V \quad k = 0, 1, 2, \dots \quad (03)$$

dont on va essayer d'estimer sa convergence ?

Nous définissons le vecteur d'erreur :

$$e^{(k)} = X_k - X^* \quad \text{associé à la } k\text{-ième itération.}$$

Comme X^* vérifie :

$$X^* = TX^* + V \quad \text{et} \quad X_k = TX_{k-1} + V.$$

En soustrayant les deux équations précédentes, on obtient :

$$e^{(k)} = Te^{(k-1)}, \quad k=0, 1, 2, \dots \quad \text{et} \quad e^{(k)} = Te^{(k-1)} = T^k e^{(0)}, \text{ donc :}$$

$$\forall e^{(0)} : \quad \lim_{k \rightarrow \infty} e^{(k)} = 0 \quad \text{ou} \quad \lim_{k \rightarrow \infty} T^k e^{(0)} = 0.$$

Ce qui revient à dire que : $\lim_{k \rightarrow \infty} T^k = 0$.

D'où la convergence de notre suite
Selon le théorème (01) [chap1].

Dans le dernier paragraphe de ce chapitre, on a choisit la méthode de "Relaxation" qui est peut être avantageuse à titre de comparaison ainsi grâce à ces critères de convergence, cités dans le chapitre suivant, qui implantent une équivalente méthode considérée éventuellement bonne, qui sera exposée au troisième chapitre.

2) Méthode de Relaxation

Son principe est basé sur la méthode de Gauss-Seidel avec quelques modifications.

On commence par factoriser la matrice A en $A = M - N$ et on pose :

$$N = U \quad \text{où} \quad M = D - L$$

D : Est la diagonale de A ($D = P^{-1}AP$)

L : Sa partie triangulaire supérieure (on pourrait aussi bien regarder son analogue triangulaire inférieur).

On renonce à exiger que pour tout i et j : $m_{i,j} = a_{i,j}$ ou $m_{i,j} = 0$

, précisément on pose : $M_\omega = \frac{D}{\omega} - L$

Cette matrice est bien triangulaire supérieure, à diagonale inversible, donc, elle est inversible, ici $\omega \neq 0$ est appelé le paramètre de relaxation.

Expression de la matrice d'itération qui sera notée T_ω .

$$A = D - L - U = \left(\frac{D}{\omega} - L\right) - N_\omega, \text{ avec}$$
$$N_\omega = \left(\frac{1-\omega}{\omega}D + U\right).$$

Cette méthode itérative consiste donc à résoudre :

$$\left(\frac{D}{\omega} - L\right) X_{K+1} = \left(\frac{1-\omega}{\omega}D + U\right) X_K + B$$

où : $X_{K+1} = T_\omega X_K + V_\omega$, telle que : $V_\omega = \left(\frac{D}{\omega} - L\right)^{-1} B$.

On note : $T_\omega = \left(\frac{D}{\omega} - L\right)^{-1} \left(\frac{1-\omega}{\omega}D + U\right)$, la matrice associée à la méthode de relaxation de paramètre ω , et si:

1- $\omega > 1$, on parle de sur-relaxation

2- $\omega < 1$, on parle de sous-relaxation.

3- On sait que la convergence de la méthode est conditionnée par : $\rho(T_\omega) < 1$, et le cas où $\omega = 1$, c'est bien la méthode de Gauss-Seidel. Donc, l'objectif général est de trouver la valeur optimale de ω , notée ω_0 :

Tel que : $\rho(T_{\omega_0}) = \inf_{\omega \in I} \rho(T_\omega)$, avec $\rho(T_{\omega_0})$ est le rayon spectral optimal

-correspondant qui est explicitement donné par :

$$\rho(T_{\omega_0}) = \omega_0 - 1 = \frac{1 - \sqrt{1 - \rho(\lambda)^2}}{1 + \sqrt{1 - \rho(\lambda)^2}}.$$

Nous terminons cette section sur un premier résultat concernant le domaine I dans lequel on peut valablement espérer de trouver un paramètre optimal de relaxation ω_0 et on peut suggérer que le système qu'on est amené à résoudre est aussi simple que dans le cas de Gauss-Seidel, puisque cela est clair à partir de l'équation :

$X_{K+1} = T_\omega X_K + V_\omega$, équivalente pour $k=1, \dots, n$ au système suivant :

$$\frac{1}{\omega} a_{1,1} X_{1,K+1} = \frac{1-\omega}{\omega} X_{1,K} - a_{1,2} X_{2,K} - \dots - a_{1,n} X_{n,K} + b_1$$

$$\frac{1}{\omega} a_{2,2} X_{2,K+1} = -a_{2,1} X_{1,K} - \frac{1-\omega}{\omega} X_{2,K} \dots - a_{2,n} X_{n,K} + b_2$$

·
·
·
·

$$\frac{1}{\omega} a_{n,n} X_{n,K+1} = -a_{n,1} X_{1,K+1} - \dots - a_{n,n-1} X_{n-1,K+1} + \frac{1-\omega}{\omega} X_{n,K} + b_n$$

La méthode de relaxation sera, donc à préférer dans toute situation où on peut déterminer une diminution de l'ordre de grandeur du rayon spectral, donc du nombre d'itération à effectuer.

2.1) Erreurs de la méthode

Nous les rappelons, les critères d'arrêt sont identiques à ceux utilisés dans la méthode de Jacobi. On note r un vecteur résidu tel que :

$$r^k = B - AX_K$$

de sorte que le critère d'arrêt soit : $\frac{\|r^k\|}{\|B\|} < \varepsilon$, avec ε une précision donnée.

Une autre technique consiste à utiliser un autre test d'arrêt basé sur :

$$\frac{\|X_K - X_{k-1}\|}{\|X_K\|} < \varepsilon$$

Lorsque l'optimum est voisin de 0, on se satisfait alors du critère d'arrêt suivant :

$$\|X_K - X_{k-1}\| < \varepsilon.$$

2.2) Algorithme

Choisir un vecteur initial X_0 , le critère d'arrêt ε et le coefficient de relaxation ω .

Créer D avec $d_{i,j}=a_{ii}$.

Créer L avec $\begin{cases} \text{pour } i > j : l_{ij} = a_{ij}, \\ \text{pour } i < j : l_{ij} = 0. \end{cases}$

Créer U avec $\begin{cases} \text{pour } i < j : u_{ij} = a_{ij}, \\ \text{pour } i > j : u_{ij} = 0. \end{cases}$

Si $\frac{\|r^k\|}{\|B\|} < \varepsilon$, $\frac{\|X_K - X_{k-1}\|}{\|X_K\|} < \varepsilon$, ou bien $\|X_K - X_{k-1}\| < \varepsilon$

ou $\varepsilon^k < \varepsilon^{k-1}$.

Calculons : $r^k = B - AX_K$

$$X_{K+1} = (D - L)^{-1} U \cdot X_K + (D - L)^{-1} B,$$

$$\tilde{X}_{k+1} = X_K + \omega (X_{K+1} - X_K).$$

Fin.

2.3) Convergence

Principe

A est une matrice inversible, pour résoudre $AX = B$, on doit traiter le problème $X = TX + V$ tout en commençant à construire l'algorithme de la méthode de Relaxation définit en général comme suit :

$$\begin{cases} X_0 & \text{Vecteur d'initialisation} \\ X_{k+1} = TX_k + V \end{cases} \quad (03)$$

où la matrice T et le vecteur V sont définis en fonction de A et B à partir d'une décomposition judicieuse de A de sorte que le rayon spectral de T soit inférieur à 1.

2.3.1) Définition (08)

La méthode itérative définie par (03) est convergente si pour toute Initialisation X_0 , on a : $\lim_{k \rightarrow \infty} X_k = x^*$, où est x^* la solution de (02)

2.3.2) Définition (09)

On dit que la méthode itérative est convergente si quel que soit le vecteur initial $X_0 \in \mathbb{R}^n$ et quel que soit le second membre $B \in \mathbb{R}^n$, la suite $(X_k)_{k \geq 0}$ définie par la méthode itérative converge vers $X = A^{-1} B$.

2.3.3) Théorème (02)

La suite définie par $X_0 \in \mathbb{R}^n$ et $X_{k+1} = TX_k + V$ converge vers $x^* = (I - T)^{-1}V$ quelque soit X_0 si et seulement si : $\rho(T) < 1$, où $\rho(T)$ est le rayon spectrale qui est en même temps considéré comme une norme équivalente à celle de la matrice T ($\rho(T) \leq c \|T\|$ où $c = \text{cte positive}$).

2.4) Convergence de la Relaxation

C'est une amélioration de celle de la méthode de Gauss-Seidel quand $\rho(T_{GS})$ est proche de 1, pour diminuer ce rayon spectral, qui gouverne la vitesse de convergence de l'algorithme.

Elle correspond à la décomposition suivante de A, ω étant un paramètre réel:

$$A = M_\omega - N_\omega \text{ avec } M_\omega = \frac{D}{\omega} - L, N_\omega = \frac{1-\omega}{\omega} D + U$$

2.4.1) Théorème (03)

Soit T_ω (ω réel), la matrice de relaxation associée à A, Alors :

$$\rho(T_\omega) \geq |\omega - 1|.$$

En particulier, si la méthode converge, nécessairement, $\omega \in]0,2[$.

2.4.2) Théorème (04) (Ostrovski-Reich)

Soit A une matrice hermitienne définie positive.

Si $\omega \in]0,2[$, la méthode de relaxation converge. En particulier, pour $\omega = 1$, on ait devant la convergence de la méthode de Gauss-Seidel

2.4.3) Théorème (05)

La méthode de relaxation peut converger que sous la condition nécessaire : $\omega \in]0,2[$.

Preuve :

Soit $\lambda_1, \dots, \lambda_n$ les valeurs propres de T_ω dans \mathbb{R} comptées avec multiplicités

$$\lambda_1 \dots \lambda_n = \det(T_\omega) = \frac{(\det \frac{1-\omega}{\omega} D + F)}{\det(\frac{D}{\omega} - E)}$$

La matrice en jeu est triangulaire, donc leurs déterminants ne dépendent que de leurs diagonales

$$\lambda_1, \dots, \lambda_n = \frac{\left(\frac{1-\omega}{\omega}\right)^n \det(D)}{\left(\frac{1}{\omega}\right)^n \det(D)} = |1 - \omega|^n$$

(on a utilisé la formule $\det(\lambda A) = \lambda^n \det(A)$ pour $\lambda \in K$)

$$\begin{aligned} \text{On a la minoration } |\rho(T_\omega)| &= \max(|\lambda_1|, \dots, |\lambda_n|) \\ &= (\max(|\lambda_1|, \dots, |\lambda_n|))^{1/n} \\ &\geq (|1 - \omega|)^{1/n} \\ &= |\omega - 1|. \end{aligned}$$

2.5) Condition de convergence

Nous cherchons les limites de ω entre lesquelles nous sommes assurés de la convergence de la méthode.

Nous étudierons brièvement trois cas :

- A matrice quelconque :

On montre que pour toute matrice A une condition nécessaire de convergence est que : $0 < \omega < 2$.

- A matrice à diagonale dominante :

Une condition suffisante de convergence est que A soit à diagonale strictement dominante avec : $0 < \omega \leq 1$.

- A matrices symétrique définie positive :

Pour une matrice symétrique, définie positive la méthode de relaxation est convergente si et seulement si : $0 < \omega < 2$.

2.6) Comparaisons

2.6.1) Proposition (01)

Une méthode itérative définie par (03) converge d'autant plus vite que le rayon spectral $\rho(T)$ est proche de 0.

2.6.2) Théorème (06)

Soit A une matrice tri-diagonale, alors les rayons spectraux des méthodes de Gauss-Seidel et de Jacobi sont reliés par :

$$\rho(T_{GS}) = [\rho(T_J)]^2$$

Ainsi, pour ce type de matrices, les deux méthodes convergent ou divergent simultanément et si elles convergent, Gauss-Seidel est la plus rapide.

2.6.3) Théorème (07)

Soit A une matrice hermitienne définie positive, tri-diagonale :

Si $\omega \in]0,2[$, les méthodes de Relaxation, Gauss-Seidel et Jacobi convergent.

Pour $\omega_0 = \frac{2}{1+\sqrt{1-[\rho(T_J)]^2}}$ de relaxation optimale, alors:

$$\rho(\omega_0) = \inf_{0 < \omega < 2} \rho(T_\omega) = \omega_0 - 1 \leq \rho(T_{GS}) \leq \rho(T_J).$$

Remarque (01)

Les conséquences de ce chapitre basées sur l'étude de la convergence de la Relaxation peuvent transformer notre problème en un problème d'optimisation représenté par des bonnes méthodes qu'on peut les aviser à titre de comparaison dans le chapitre suivant.

Chapitre 3

Méthode du Gradient

Ce troisième chapitre montre l'existence d'un tel isomorphisme entre deux méthodes, signalé par des critiques sur la convergence de la Relaxation, qui ont transformées notre problème en un problème d'optimisation représenté par des méthodes de minimisations, tel que l'essai de celle du gradient est avérée excellente.

1) Principe

L'idée principale derrière ce chapitre est de transformer notre problème, qui est la résolution du système :

$$AX = B \quad (04)$$

en un Problème d'optimisation de la forme :

$$F(X) = \min_{x \in V} F(x) \quad (05)$$

Pour une certaine fonction F qui reste à préciser, définie sur l'espace vectoriel V .

Considérons une matrice symétrique et définie-positive. On pose :

$$F(X) = \frac{1}{2}(AX, X) - (B, X) \quad (06)$$

La condition d'optimalité du problème (06) est donnée par :

$$\nabla F(X) = 0.$$

Ici on rappelle que la dérivée directionnelle s'évalue à l'aide de la formule :

$$(\nabla F(X), X) = \frac{d}{d\varepsilon} \frac{F(X+\varepsilon)}{\varepsilon} = 0.$$

Donc de la formule (06), on aura :

$$\nabla F(X) = AX - B = -r \quad (r \text{ est le résidu}).$$

1.1) Théorème (08) :

Soit le système $AX = B$ où $A = (a_{ij})_{n \times n}$ est une matrice définie positive, alors résoudre $AX = B$ revient à trouver le point $X \in \mathbb{R}^n$ où

$$F(V) = \frac{1}{2}(AV, V) - (B, V), \text{ atteint son minimum, } v \in \mathbb{R}^n.$$

Démonstration

Calculons $F(V) - F(X)$, on a :

$$\begin{aligned} F(V) - F(X) &= \frac{1}{2}(AV, V) - (B, V) - \left(\frac{1}{2}(AX, X) - (B, X) \right) \\ &= \frac{1}{2}(AV, V) - (B, V) + \frac{1}{2}(AX, X) \end{aligned} \quad (07)$$

On a: $V^T AX = \frac{1}{2} V^T AX + \frac{1}{2} V^T AX$, portons l'expression de $V^T AX$ dans (07), nous obtenons :

$$\begin{aligned} F(V) - F(X) &= \frac{1}{2} V^T AV - \frac{1}{2} V^T AX - \frac{1}{2} V^T AX + \frac{1}{2} X^T AX \\ &= \frac{1}{2} V^T A(V-X) - \frac{1}{2} (V-X)^T AX \end{aligned} \quad (08)$$

Posons : $\xi = V - X$, L'équation (08) devient :

$$F(V) - F(X) = \frac{1}{2} V^T A \xi - \frac{1}{2} \xi^T AX \quad (\xi^T AX \in \mathbb{R}^n)$$

Car $((\xi^T AX)^T)^T = \xi^T AX$
 $= X^T A \xi$ et $A^T = A$

$$F(V) - F(X) = \frac{1}{2} \xi^T A \xi > 0, \text{ pour tout } V \neq X$$

$F(V) > F(X)$ Pour tout $V \neq X$, donc X est le point où F atteint son minimum.

2) Méthode de descente (Gradient)

Les méthodes qu'on va développer pour résoudre numériquement les systèmes (01), où A est définie positive, reposent sur cette propriété des systèmes définies positifs.

ces méthodes appartiennent à la classe des méthodes itératives. Elles consistent à créer une suite de vecteurs $\{X_i \in \mathbb{R}^n\}$ telle que : $X_{i+1} = X_i + \mu v$, $v \in \mathbb{R}^n$, $\mu \in \mathbb{R}$ et la fonction $F(X_{i+1})$ ait la plus petite valeur possible (un minimum).

Cherchons donc cette valeur.

On a :

$$F(X_{i+1}) = \frac{1}{2} X_{i+1}^T A X_{i+1} - X_{i+1}^T B \quad / \quad X_{i+1} = X_i + \mu v$$

$$F(X_{i+1}) = \left(\frac{1}{2} v^T A v\right) \mu^2 - v^T (B - A X^{(i)}) \mu + \frac{1}{2} X_i^T A X_i - X_i^T B$$

Posons : $\mathcal{R}^{(i)} = B - A X_i$, Alors :

$$F(X_{i+1}) = \left(\frac{1}{2} v^T A v\right) \mu^2 - v^T \mathcal{R}^{(i)} \mu + F(X_i) \quad (09)$$

L'équation (09) est un trinôme en μ , il atteint son minimum au point μ vérifiant l'équation :

$$\mu (v^T A v) - v^T \mathcal{R}^{(i)} = 0$$

$\Rightarrow \mu = \frac{v^T \mathcal{R}^{(i)}}{v^T A v}$, donc notre processus itératif pour la résolution de (4.1) définie par :

$$X^{(i+1)} = X^{(i)} + \mu v, \quad v \in \mathbb{R}^n$$

et

$$\mu = \frac{v^T \mathcal{R}^{(i)}}{v^T A v}, \quad \text{où } \mathcal{R}^{(i)} = B - A X^{(i)}.$$

2.2) Remarque (02)

v est un vecteur arbitraire, alors dans ce qui suit, nous allons construire des méthodes selon le choix de v .

Par exemple si $v = -(\text{Grad } F)(X^{(i)})$ où $\text{Grad } F =$ gradient de de la fonction quadrature F , on obtient la méthode de la plus grande pente :

$$X^{(i+1)} = X^{(i)} - \mu(\text{Grad } F)(X^{(i)})$$

avec

$$\mu = \frac{v^T \mathcal{R}^{(i)}}{v^T A v}$$

2.3) Théorème (09)

Soit $AX = B$, alors le résidu $\mathcal{R}^{(i)} = B - AX^{(i)}$ est égale à : $-(\text{Grad } F)(X^{(i)})$
 où $F(X) = \frac{1}{2}(AX, X) - (B, X)$ et $\mathcal{R}^{(i+1)} \perp v$.

Démonstration :

On a :

$$F(X) = \frac{1}{2} \sum \sum x_i a_{ij} x_j - \sum x_i b_i \quad \text{où } X = (x_1, \dots, x_n)^T \text{ et } b = (b_1, \dots, b_n)^T$$

$$\frac{\partial F}{\partial x_k} = \frac{\partial}{\partial x_k} \left(\frac{1}{2} \sum \sum x_i a_{ij} x_j - \sum x_i b_i \right), \quad k=1, 2, \dots, n$$

$$= \frac{\partial}{\partial x_k} \left(\frac{1}{2} \sum \sum x_i a_{ij} x_j \right) - \frac{\partial}{\partial x_k} \sum x_i b_i, \quad k=1, 2, \dots, n$$

$$= \left(\frac{1}{2} \sum \sum x_i a_{ij} x_j \right) - b_k, \quad k=1, \dots, n$$

$$= \frac{1}{2} (\sum a_{kj} x_j + \sum a_{ik} x_i) - b_k, \quad k=1, 2, \dots, n$$

$$\sum a_{ik} x_i - b_k = \sum a_{kj} x_j - b_k, \quad k=1, 2, \dots, n$$

$$\begin{aligned} \text{Car } A^T &= A \\ &= \sum a_{ik} x_i - b_k \quad / \quad k=1, 2, \dots, n \end{aligned}$$

ce système de n équations peut être mis sous la forme vectorielle suivante:

$$\begin{pmatrix} \frac{\partial F}{\partial x_1} \\ \frac{\partial F}{\partial x_n} \end{pmatrix} = AX - B \implies (\text{Grad } F)(X_i) = AX_i - B = -R^{(i)}$$

Démontrons que $R^{(i+1)} \perp V$? Calculons $V^t R^{(i+1)}$.

$$\begin{aligned} \text{On a ; } V^t R^{(i+1)} &= V^t (B - AX^{(i+1)}) \\ &= V^t (B - A(X^{(i)} + \mu v)) = V^t R^{(i)} - \frac{v^T \mathcal{R}^{(i)}}{v^T A v} \\ &= V^t R^{(i)} - V^t R^{(i)} = 0 \end{aligned}$$

$V^t R^{(i+1)} = 0$. Donc $R^{(i+1)} \perp V$.

Avec ce résultat, la méthode de la plus grande pente devient :

$$\begin{aligned} X^{(i+1)} &= X^{(i)} + \mu R^{(i)} \\ \text{avec } \mu &= \frac{(\mathcal{R}^{(i)})^T \mathcal{R}^{(i)}}{(\mathcal{R}^{(i)})^T A \mathcal{R}^{(i)}}. \end{aligned}$$

2.4 Convergence de la méthode

Pour simplifier les calculs, on peut garder μ constant. Cependant pour assurer la convergence de la méthode, La valeurs μ doit satisfaire l'inégalité suivante :

$$\mu < \frac{2}{\lambda_{max}}, \quad \text{où } \lambda_{max} \text{ est la grande valeur propre de la matrice } A.$$

En effet,

$$\mathcal{R}^{(i+1)} = B - AX^{(i+1)} = B - A(X^{(i)} + \mu \mathcal{R}^{(i)})$$

$$= B - AX^{(i)} + \mu A \mathcal{R}^{(i)} = \mathcal{R}^{(i)} - \mu A \mathcal{R}^{(i)} = (I - \mu A) \mathcal{R}^{(i)}$$

$$\mathcal{R}^{(i+1)} = (I - \mu A)^2 \mathcal{R}^{(i-1)} = \dots = (I - \mu A)^{i+1} \mathcal{R}^{(0)}$$

ou bien $\mathcal{R}^{(i)} = (I - \mu A)^i \mathcal{R}^{(0)}$.

Puisque A est définie positive, elle possède donc n vecteurs propre linéairement indépendants.

Soient Z_1, \dots, Z_n ces vecteurs propres. Donc $\mathcal{R}^{(0)} \in \mathbb{R}^n$ peut s'écrire :

$$\mathcal{R}^{(0)} = a_1 Z_1 + \dots + a_n Z_n$$

posons $y_k = a_k Z_k$ pour $k=1, \dots, n$, alors :

$$\mathcal{R}^{(0)} = y_1 + y_2 + \dots + y_n \text{ et}$$

$$\mathcal{R}^{(i)} = (I - \mu A)^i \sum_{k=1}^n y_k = (I - \mu A)^{(i-1)} (\sum_{k=1}^n (1 - \mu \lambda_k) y_k), \text{ où } \lambda_k \text{ est la}$$

valeur propre associée au vecteurs propre $z_k = \frac{y_k}{a_k}$, où y_k est aussi vecteur propre associé à la valeur propre λ_k .

De la même manière, on a :

$$\mathcal{R}^{(i)} = \sum_{k=1}^n (1 - \mu \lambda_k)^i y_k = \sum_{k=1}^n C_k^i y_k, \text{ où } C_k = 1 - \mu \lambda_k$$

On constate que : $\mathcal{R}^{(i)} \rightarrow 0$, lorsque $i \rightarrow \infty$, si $|C_k| < 1$, c'est-à-dire si :

$$|C_k| = |1 - \mu \lambda_k| < 1 \Leftrightarrow -1 < 1 - \mu \lambda_k < 1 \Rightarrow \mu \lambda_k < 2 \Rightarrow \lambda_k < \frac{2}{\mu}; k = 1, \dots, n.$$

Donc si $\lambda_{max} < \frac{2}{\mu}$ le processus itératifs converge.

Exemple d'Application :

Soit le système linéaire creux suivant :

$$Ax = b \dots\dots\dots (01)$$

Tell que :

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix}, \quad A \in M_{3,3}(\mathbb{R})$$

$$b = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}, \quad b \in \mathbb{R}^3$$

Réolvons ce système pour $x_0 = 0_{\mathbb{R}^3} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$?

Tout d'abord vérifions que A est définie positive ? On commençant par calculer ces valeurs propres:

$$|A - \mu I| = 0$$

$$\begin{vmatrix} 1 - \mu & 0 & 0 \\ 0 & 1 - \mu & 0 \\ 0 & 0 & 2 - \mu \end{vmatrix} = 0$$

$\Rightarrow (1-\mu)[(1-\mu).(2-\mu)] = 0 \Rightarrow \mu = 1$ et $\mu = 2$, toutes les deux sont positives, alors A est définie positive.

Méthode de gradient

La convergence de la méthode de relaxation a créé un isomorphisme qui transforme le problème algébrique (01) à un Problème d'optimisation de la forme :

$$F(X) = \min_{x \in V} F(x) \dots\dots\dots (02)$$

On a :

$$F(x) = \frac{1}{2}(Ax, x) - (b, x) = F(x) = x^t Ax - x^t b$$

$$F(x) = \frac{1}{2} (x \ y \ z) \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} - (x \ y \ z) \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}$$

Alors :

$$F(x) = \frac{1}{2}x^2 + \frac{1}{2}y^2 + z^2 - x - 2y$$

Calculer le gradient de la forme $F(x)$:

$$\begin{aligned} \frac{\nabla F}{dx} &= x - 1 \\ \frac{\nabla F}{dy} &= y - 2 \\ \frac{\nabla F}{dz} &= 2z \end{aligned}$$

$$\text{On a : } \nabla F(0 \ 0 \ 0) = \begin{pmatrix} -1 \\ -2 \\ 0 \end{pmatrix} \neq \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

Alors le point x_0 n'est pas le minimum de (02), on va le chercher comme suit :

On a :

$$X_{i+1} = X_i + \mu v \quad \text{et} \quad F(X_{i+1}) = \frac{1}{2} X_{i+1}^T A X_{i+1} - X_{i+1}^T b$$

On a :

$$x_1 = x_0 + \mu v$$

$$V = -\nabla F(x)$$

$$x_1 = x_0 - \mu \nabla F(x_0)$$

$$x_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} -\mu \\ -2\mu \\ 0 \end{pmatrix}$$

$$x_1 = \begin{pmatrix} \mu \\ 2\mu \\ 0 \end{pmatrix}$$

$$F(x_1) = \frac{5}{2}\mu^2 - 5\mu$$

Ecrire la forme quadratique $\varphi(\mu)$:

$$\varphi'(\mu) = 0$$

$$\varphi'(\mu) = 5\mu - 5 = 0 \Rightarrow \mu = 1$$

La direction de descente : $V = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}$, alors : $x_1 = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix}$

$$\frac{\nabla F(x_1)}{dx} = (1) - 1 = 0$$

$$\frac{\nabla F(x_1)}{dy} = (2) - 2 = 0$$

$$\frac{\nabla F(x_1)}{dz} = 2(0) = 0$$

Alors $\nabla F(x_1) = 0$

Donc x_1 est le minimum de (02), ce qui induit qu'il est la solution de (01).

Conclusion

Notre mémoire est quasiment consacré sur l'application de la méthode itérative de Relaxation qui a établie un isomorphisme remarquable transformant notre étude en un problème d'optimisation, justifiant la problématique de notre thème, aussi considérée comme source de plusieurs méthodes de minimisations engendrées par la fameuse suite de vecteurs $\{X_i \in \mathbb{R}^n\}$:

$$X_{i+1} = X_i + \mu v, \quad \text{où } v \in \mathbb{R}^n, \mu \in \mathbb{R}.$$

Ces méthodes sont constructives et techniques (génératrices) selon le vecteur $v \in \mathbb{R}^n$, tel que à titre d'exemple : si on fixe v par : $(\text{Grad } F)(X_i)$, on obtient la méthode de la plus grande pente (Descente) qui est avérée bonne !

Remarques et perspectives

- 1- Les méthodes directes de résolution (basés sur une factorisation) sont appréciés pour leur grande stabilité numérique mais ont des besoins mémoire importants à l'inverse des méthodes itératives qui peuvent être peut consommatrice en mémoire mais pas nécessairement fermes numériquement.
- 2- Les matrices creuses, telles que les matrices tridiagonales sont structurées et le problème de stockage ne se pose pas, l'accès à un élément est identifié à l'aide d'une fonction d'accès, alors le problème des matrices creuses apparait dans le cas où les zéros sont répartis d'une manière aléatoire.
- 3- Suite à cette dernière remarque et ce qu'on a cité comme aidé sur la procédure du stockage des matrices creuses dans la deuxième partie du chapitre 01, on suggère une étude (algorithme) perspective à cette effet, afin améliorer en plus l'approximation des systèmes creux.

Bibliographies

- [01] CONTE.S.D, DE BOOR.C: "Elementary Numerical Analysis. An Algorithmic Approach", ISE 1981,
- [02] DEMAILLY.J.P : "Analyse Numérique et Équations Différentielles", OPU ALGER,
- [03] DEMIDOVITCH.B et MARON.I : "Éléments De Calcul Numérique", Moscou 1979,
- [04] EMMANUEL AGULLO : " Résolution Out-Of-Core De Systèmes Linéaires Creux De Grand Taille", Lyon 2005, N^o 5668,
- [05] HENRICI.P, WILLEY JOHN: "Elements of Numerical Analysis", SONS 1967,
- [06] JEREMIE GAIDAMOUR : "Thèse Conception D'un Solveur Linéaire Creux Parallèle Hybride Direct- Iterative ", N^o 3904,
- [07] LEGRAS.J : "Précis D'analyse Numérique", Paris 1963,
- [08] MANFRED GILLI : "METHODES NUMERIQUES (2006)",
- [09] RAIRO : " Modélisation Mathématique et Analyse Numérique - Tome 27- N₀ 6-1993",
- [10] SIBONY.M et MARDON.J.CI : "Analyse Numérique", Tomes 1 et 2, Paris 1984,
- [11] THEODOR.R : "Initiation A L'analyse Numérique", Paris 1989.