Chapitre 4

Méthodes itératives de résolution des systèmes linéaires

4.1 Généralités

On se donne une matrice inversible A et un système linéaire

$$Au = b. \tag{4.1}$$

On désire transformer (4.1) en un système équivalent sous la forme d'un problème de point-fixe

$$u = Bu + c \tag{4.2}$$

avec l'hypothèse que $I - B$ est inversible. Pour calculer le point-fixe de (4.2), on dispose de l'algorithme bien connue

$$\begin{cases} u_0 & \text{donné,} \\ u_{k+1} = & Bu_k + c. \end{cases}$$

Pour étudier la convergence de cet algorithme, il suffit de considérer la suite des erreurs $e_k = u_k - u$ avec $e_0 = u_0 - u$ et de prouver que

$$e_k \underset{k \to \infty}{\longrightarrow} 0.$$

On a le résultat suivant comme critère fondamental de convergence.

Théorème 15 *Les énoncés suivants sont équivalents :*

1. *la méthode itérative (4.2) converge pour tous les* $u_0 \in \mathbb{R}^n$,
2. *le rayon spectral de* B *est inférieur à un, i.e.* $\rho(B) < 1$,
3. *pour au moins une norme matricielle subordonnée, on a que* $\|B\| < 1$.

Preuve :

$$\begin{array}{rr} & u_{k+1} = Bu_k + c \\ - & u = Bu + c \\ \hline & e_{k+1} = Be_k \end{array}$$

Ceci implique que

$$e_k = B^k e_0$$

Ainsi nous sommes conduit à étudier la suite des itérés B^k. Ceci nous ramène au théorème du chapitre 1 (4).

L'algorithme ci-dessus n'est nul autre que la méthode de Picard pour le calcul du point-fixe de l'application contractante

$$F : \mathbb{R}^n \longrightarrow \mathbb{R}^n$$

définie par

$$F(v) = Bv + c$$

En effet, F vérifie l'inégalité

$$\|F(u) - F(v)\| \leq \|B\| \|u - v\|$$

en prenant la norme matricielle du théorème 15. Or $\|B\| < 1$, ce qui montre que F est une application contractante.

Examinons le comportement asymptotique de l'erreur. Pour simplifier, supposons que B est une matrice normale.

$$\begin{aligned} & \|e_k\|_2 = \|B^k e_0\|_2 \leq \|B\|^k \|e_0\|_2 \\ \Longrightarrow \quad & \|e_k\|_2 \leq \rho(B)^k \|e_0\|_2 \end{aligned}$$

Ainsi la suite $\|e_k\|_2$ se comporte asymptotiquement *au pire* comme la suite géométrique $\rho(B)^k$. Grâce à cette identification, on peut répondre à deux questions sur la convergence des méthodes itératives.

– Etant donné une méthode itérative, déterminer si elle converge. Il suffit de vérifier que

$$\rho(B) < 1 \text{ ou } \|B\| < 1$$

– Etant donné deux méthodes itératives convergentes, comparer les vitesses de convergence. Il suffit de comparer les rayons spectraux ρ_1 et ρ_2 :

$$\rho_1 < \rho_2 \qquad \Longrightarrow \qquad \text{la } 1^{\text{re}}\text{méthode est plus rapide que la } 2^{\text{e}}.$$

4.2 Méthodes de Jacobi, de Gauss-Seidel et de relaxation

Introduisons la partition $A = D - E - F$ où

$$\begin{cases} D_{ij} & = & a_{ij}\delta_{ij} \\ E_{ij} & = & \begin{cases} -a_{ij} & i > j \\ 0 & i \leq j \end{cases} \\ F_{ij} & = & \begin{cases} -a_{ij} & i < j \\ 0 & i \geq j \end{cases} \end{cases}$$

$$A = \begin{pmatrix} & & -F \\ & D & \\ -E & & \end{pmatrix}$$

Méthode de Jacobi

On fait l'hypothèse que D est inversible, i.e. $a_{ii} \neq 0 \; \forall i$. On écrit $Au = b$ sous la forme

$$\begin{aligned} & Du = (E+F)u + b \\ \Longrightarrow \qquad & u = D^{-1}(E+F)u + D^{-1}b \end{aligned}$$

qui est de la forme $u = Bu + c$ si on choisit

$$\begin{aligned} B &= D^{-1}(E+F) \\ c &= D^{-1}b \end{aligned}$$

Algorithme de Jacobi

$$\begin{cases} u^{(0)} & \text{donné} \\ u_i^{(k+1)} = \dfrac{b_i - \sum_{j \neq i} a_{ij} u_j^{(k)}}{a_{ii}} & \forall i = 1, \ldots, n. \end{cases}$$

Il existe aussi une variante pour les matrices partitionnées par blocs.

Méthode de Gauss-Seidel

On fait la même hypothèse que pour la méthode de Jacobi, c'est-à-dire que D soit inversible i.e. $a_{ii} \neq 0 \; \forall i$. Ceci implique que $D - E$ est aussi inversible.

On écrit maintenant $Au = b$ sous la forme

$$\begin{aligned} & (D-E)u = Fu + b \\ \Longrightarrow \qquad & u = (D-E)^{-1}Fu + (D-E)^{-1}b \end{aligned}$$

qui est de la forme $u = Bu + c$ si on choisit

$$\begin{aligned} B &= (D-E)^{-1}F \\ c &= (D-E)^{-1}b \end{aligned}$$

Algorithme de Gauss-Seidel

$$\begin{cases} u^{(0)} & \text{donné} \\ u_i^{(k+1)} = \dfrac{b_i - \sum_{j<i} a_{ij} u_j^{(k+1)} - \sum_{j>i} a_{ij} u_j^{(k)}}{a_{ii}} & \forall i = 1, \ldots, n. \end{cases}$$

Il existe aussi une variante pour les matrices partitionnées par blocs.

Méthode de relaxation

On fait la même hypothèse que pour les autres méthodes, 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 $Au = b$ sous la forme générale

$$\begin{aligned} &(\frac{D}{\omega} - E)u = (F + \frac{1-\omega}{\omega} D)u + b \\ \Longrightarrow \qquad & u = (\frac{D}{\omega} - E)^{-1}(F + \frac{1-\omega}{\omega} D)u + (\frac{D}{\omega} - E)^{-1} b \end{aligned}$$

qui est de la forme $u = Bu + c$ si on choisit

$$\begin{aligned} B &= (\frac{D}{\omega} - E)^{-1} \left(F + \frac{1-\omega}{\omega} D \right) \\ c &= (\frac{D}{\omega} - E)^{-1} b \end{aligned}$$

Algorithme de relaxation

$$\begin{cases} u^{(0)} & \text{donné} \\ u_i^{(k+1)} = (1-\omega) u_i^{(k)} + \dfrac{\omega}{a_{ii}} \left[b_i - \sum_{j<i} a_{ij} u_j^{(k+1)} - \sum_{j>i} a_{ij} u_j^{(k)} \right] & \forall i = 1, \ldots, n. \end{cases}$$

Ceci revient à calculer la valeur fournie par l'algorithme de Gauss-Seidel et de faire une combinaison linéaire de cette valeur avec l'itéré précédent, ce qui constitue l'étape de relaxation.

Pour $i = 1, \ldots, n$ faire :

1. calculer $\tilde{u}_i$ par Gauss-Seidel

$$\tilde{u}_i^{(k+1)} = \frac{b_i - \sum_{j<i} a_{ij} u_j^{(k+1)} - \sum_{j>i} a_{ij} u_j^{(k)}}{a_{ii}}$$

2. relaxer cette valeur

$$u_i^{(k+1)} = u_i^{(k)} + \omega(\tilde{u}_i^{(k+1)} - u_i^{(k)})$$

4.3 Etude de la convergence des méthodes itératives

Nous appliquerons le résultat suivant.

Théorème 16 *Soit A une matrice hermitienne et définie-positive possédant une décomposition de la forme $A = M - N$ avec M inversible. Si $M^* + N$ est une matrice définie-positive, alors $\rho(M^{-1}N) < 1$.*

Preuve : On a que $M^* + N$ est hermitienne car

$$(M^* + N)^* = M + N^*$$

$$\text{Or sachant que } A^* = A \qquad \Longrightarrow \qquad N^* = M^* - M + N$$

$$\Longrightarrow \qquad M + N^* = M + M^* - M + N = M^* + N$$

Pour prouver que $\rho(M^{-1}N) < 1$, il suffit de montrer, grâce au théorème (15), que $\|M^{-1}N\| < 1$ pour une certaine norme matricielle subordonnée.

Choisissons la norme matricielle $\|v\|_A = \sqrt{(Av, v)}$ car A est définie positive et hermitienne.

$$\|M^{-1}N\|_A = \|I - M^{-1}A\|_A = \max_{\|v\|_A=1} \|v - w\|_A$$

avec $w = M^{-1}Av$.

Evaluons $\|v - w\|_A$:

$$\begin{aligned}\|v-w\|_A^2 &= \|v\|_A^2 + \|w\|_A^2 - (v,w)_A - (w,v)_A \\ &= \|v\|_A^2 + (Aw,w) - (Av,w) - (Aw,v) \\ &= 1 + (Mw,w) - (Nw,w) - (Mw,w) - (w,Mw) \\ &= 1 - (N + M^*\ w, w) < 1 \qquad \text{car } (N + M^*\ w, w) > 0\end{aligned}$$

Par compacité de la sphère unité $\{v \mid \|v\|_A = 1\}$, on obtient que

$$\max_{\substack{\|v\|=1 \\ w = M^{-1}Av}} \|v - w\|_A < 1$$

ce qui montre que $\rho(M^{-1}N) < 1$.

Voici un critère simple assurant la convergence de la méthode de relaxation et en particulier celle de Gauss-Seidel.

Théorème 17 *Si A est une matrice hermitienne et définie-positive, alors la méthode de relaxation converge pour toutes les valeurs du paramètre de relaxation comprises dans l'intervalle $\omega \in]0, 2[$.*

Preuve : On décompose A sous la forme

$$\begin{aligned} A = M - N = (\frac{D}{\omega} - E) - (\frac{1-\omega}{\omega} D + F) \\ \Longrightarrow \qquad M^* + N = \frac{D}{\omega} - E^* + \frac{1-\omega}{\omega} D + F\end{aligned}$$

Or $D = D^*$ et $E^* = F$. Ce qui entraîne que

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

qui est définie positive si $0 < \omega < 2$. On applique le théorème précédent (16) pour conclure que

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

et que la méthode de relaxation converge grâce au théorème (15).

On peut montrer que la condition $\omega \in]0,2[$ est aussi nécessaire même si A n'est pas définie positive.

Par la suite, on notera la matrice d'itération de la méthode de relaxation

$$\mathcal{L}_\omega = (\frac{D}{\omega} - E)^{-1}(\frac{1-\omega}{\omega}D + F)$$

Théorème 18 *On a toujours que*

$$\rho(\mathcal{L}_\omega) \geq |\omega - 1| \qquad\qquad \forall \omega \neq 0.$$

Preuve : Fixons $\omega \neq 0$. On notera les valeurs propres de $\mathcal{L}_\omega$ par λ_i.

$$\begin{aligned}\prod_{i=1}^{n} \lambda_i = \det \mathcal{L}_\omega &= \frac{\det(\frac{1-\omega}{\omega}D + F)}{\det(\frac{D}{\omega} - E)} \\ &= \frac{(\frac{1-\omega}{\omega})^n \det D}{(\frac{1}{\omega})^n \det D} \\ &= (1-\omega)^n\end{aligned}$$

Mais

$$[\rho(\mathcal{L}_\omega)]^n \geq |\prod_{i=1}^{n} \lambda_i| = |1-\omega|^n$$

ce qui entraîne que

$$\rho(\mathcal{L}_\omega) \geq |1-\omega|.$$

Maintenant comparons les différentes méthodes itératives. Commençons par comparer la méthode de Jacobi avec celle de Gauss-Seidel dans le cas où la matrice est tridiagonale.

Théorème 19 *Soit A une matrice tridiagonale. On a que*

$$\rho(\mathcal{L}_1) = (\rho(J))^2$$

où $J = D^{-1}(E+F)$ est la matrice d'itération de la méthode de Jacobi et $\mathcal{L}_1$ celle de Gauss-Seidel.

Preuve : consulter le livre de Ciarlet [1].

Autrement dit, la méthode de Jacobi converge si et seulement si la méthode de Gauss-Seidel converge. De plus, si les deux méthodes sont convergentes, la méthode de Gauss-Seidel converge plus rapidement que celle de Jacobi.

Finalement, voici le résultat de comparaison entre les méthodes de Jacobi et relaxation.

Théorème 20 *Soit A une matrice tridiagonale telle que $sp(J) \subset \mathbb{R}$. On a que :*

- *la méthode de Jacobi converge si et seulement si la méthode de relaxation converge pour tout choix de $\omega \in]0, 2[$,*
- *le graphe de la fonction $\omega \mapsto \rho(\mathcal{L}_\omega)$ a l'allure suivante*
  Le choix optimal de ω est donnée par

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

Preuve : on consultera de nouveau Ciarlet [1].

4.4 Méthodes de gradient pour la résolution des systèmes linéaires

L'idée principale derrière ces méthodes consiste à transformer le problème de calculer l'unique solution du système linéaire

$$Au = b \tag{4.3}$$

en un problème d'optimisation de la forme

$$J(u) = \min_{v \in V} J(v) \tag{4.4}$$

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

Considérons une matrice symétrique et définie-positive. On pose

$$J(v) = \frac{1}{2}(Av, v) - (b, v) \tag{4.5}$$

La condition d'optimalité du problème (4.4) est donnée par

$$\nabla J(u) = 0.$$

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

$$(\nabla J(u), v) = \frac{d}{d\epsilon} J(u + \epsilon v)_{|\epsilon=0}.$$

Avec le choix (4.5) de J ci-dessus, on obtient

$$\nabla J(u) = Au - b \equiv -r \qquad (r \text{ est le résidu})$$

Remarque : au lieu du J ci-dessus, on peut aussi prendre la fonctionnelle

$$J(v) = \frac{1}{2}(A(v-u), v-u) \equiv \frac{1}{2}\|v-u\|_A$$

sachant que $Au = b$. On peut montrer que les deux fonctionnelles diffèrent que par une constante.

Passons à la description des algorithmes de descentes pour résoudre le problème (4.4) avec J fournie par (4.5).

Il s'agit de méthodes itératives dont le prototype s'écrit :

1. choisir un vecteur initial u_0,
2. trouver une direction de descente $d_k \neq 0$ à chaque itération,
3. mettre à jour la solution u_{k+1} à l'aide le la formule

   $$u_{k+1} = u_k + \rho(u_k, d_k)d_k$$

   où $\rho(u_k, d_k)$ est un nombre réel positif dépendant de la solution et de la direction de descente à l'itération précédente.

Comme choix de direction de descente, il semble naturel de choisir

$$d_k = -\nabla J(u_k) = b - Au_k$$

Algorithme de gradient :

1. u_0 donné
2. Pour $k = 1, 2, \ldots,$ jusqu'à convergence : faire

$$u_{k+1} = u_k + \rho_k(b - Au_k)$$

Si $\rho_k \equiv \rho$, on dit que l'algorithme est à pas fixe.

Voici le résultat principal de convergence de cette classe de méthodes.

Théorème 21 *Convergence de la méthode du gradient*

Soit A une matrice symétrique et définie-positive. Si on choisit le paramètre ρ_k dans l'intervalle

$$\tilde{a} \leq \rho_k \leq \tilde{b}$$

avec les restrictions $\tilde{a} > 0$ et $\tilde{b} < \dfrac{2}{\lambda_n}$ où λ_n dénote la plus grande valeur propre de A, la méthode de gradient (à pas variable) converge et de plus il existe une constante $\beta < 1$ telle que

$$\|u_k - u\| \leq \beta^k \|u_0 - u\|$$

où u est la solution du problème $Au = b$.

Preuve : on a que

$$\begin{aligned} u_{k+1} - u &= (u_k - u) - \rho_k A(u_k - u) \\ &= (I - \rho_k A)(u_k - u) \\ \implies \qquad \|u_{k+1} - u\|_2 &\leq \|I - \rho_k A\|_2 \|u_k - u\|_2 \end{aligned}$$

Or

$$\|I - \rho_k A\|_2 = \rho(I - \rho_k A) = \max\{|1 - \rho_k\lambda_1|, |1 - \rho_k\lambda_n|\}$$

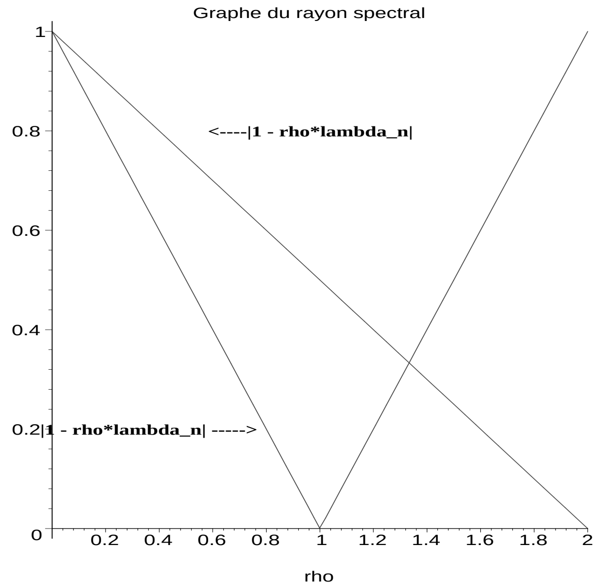

FIG. 4.1 – Graphe de la fonction $\|I - \rho A\|_2$

Considérons le graphe de la fonction $\rho \mapsto \|I - \rho A\|_2$

Prenons $\tilde{a} > 0$ et $\tilde{b} < \dfrac{2}{\lambda_n}$. Il est clair selon la figure ci-contre que si on choisit ρ_k de sorte que $\tilde{a} \leq \rho_k \leq \tilde{b}$, on aura l'inégalité

$$\|I - \rho_k A\|_2 \leq \beta < 1$$

Donc

$$\begin{aligned} & \|u_{k+1} - u\|_2 \leq \beta \|u_k - u\|_2 \\ \Longrightarrow \qquad & \|u_{k+1} - u\|_2 \leq \beta^k \|u_0 - u\|_2 \underset{k\to\infty}{\longrightarrow} 0 \end{aligned}$$

c'est-à-dire que

$$u_k \underset{k\to\infty}{\longrightarrow} u$$

Remarque : selon la figure ci-dessus, la valeur optimale du paramètre ρ de la méthode du gradient à pas fixe est fournie par

$$\rho = \frac{2}{\lambda_1 + \lambda_n}.$$

Passons à la méthode de gradient à pas optimal. L'idée de la méthode consiste à choisir ρ_k de manière optimale à chaque itération , c'est-à-dire la valeur de ρ qui réalise le minimum de la fonctionnelle J le long de la droite $u_k - \rho\nabla J(u_k)$.

Voici les étapes de l'algorithme de gradient à pas optimal.

1. u_0 donné
2. Pour $k = 1, 2, \dots,$ jusqu'à convergence : faire
   (a) on détermine ρ_k de sorte que
   $$J(u_k - \rho_k \nabla J(u_k)) = \min_{\rho} J(u_k - \rho\nabla J(u_k))$$
   (b) mise à jour de la solution
   $$u_{k+1} = u_k - \rho_k \nabla J(u_k)$$

En pratique, comment évalue-t-on le paramètre ρ_k ?

On a que
$$\frac{d}{d\rho} J(u_k - \rho\nabla J(u_k))_{|\rho=\rho_k} = 0 \Longleftrightarrow (\nabla J(u_{k+1}), \nabla J(u_k)) = 0$$
Posons $d_k = -\nabla J(u_k) = b - Au_k$. La mise à jour s'écrit $u_{k+1} = u_k + \rho_k d_k$.

Or

$$d_{k+1} = -\nabla J(u_{k+1}) = b - Au_{k+1}$$
$$d_{k+1} = b - A(u_k + \rho_k d_k)$$
$$d_{k+1} = d_k - \rho_k A d_k$$
$$\Longrightarrow \quad (d_{k+1}, d_k) = 0 \quad \Longleftrightarrow \quad (d_k - \rho_k A d_k, d_k) = 0$$
$$\Longrightarrow \quad \rho_k = \frac{\|d_k\|^2}{(Ad_k, d_k)}$$

Algorithme de gradient à pas optimal :

1. u_0 donné
2. Pour $k = 1, 2, \dots,$ jusqu'à convergence : faire

(a) on calcule le gradient $w_k = Au_k - b$

(b) on évalue $\rho_k = \dfrac{\|w_k\|^2}{(Aw_k, w_k)}$

(c) on pose $u_{k+1} = u_k - \rho_k w_k$

On peut montrer que cette méthode converge si A est définie-positive et symétrique, consulter Ciarlet [1].

Remarques :

- C'est une méthode intéressante si le calcul de Aw est facile à faire.
- Si les courbes de niveaux de J sont des cercles, alors l'algorithme converge en une seule itération car la solution se trouve sur la droite $u_0 - \rho\nabla J(u_0)$.
- Par contre si les courbes de niveau sont des ellipses très allongées, l'algorithme devient lent.
- Il n'est pas nécessaire de connaître explicitement la matrice A mais seulement le produit matrice-vecteur Av.

4.5 méthode du gradient conjugué

Soit A une matrice symétrique et définie-positive. On notera le produit scalaire associé à A par

$$(u, v)_A = (Au, v)$$